2012

Characterization Of Online Archives Of Astronomical Imaging Vis-a-vis Serendipitous Asteroids, And Their Astrometric Properties

Jean Marc Denis
University of Central Florida

CHARACTERIZATION OF ONLINE ARCHIVES OF ASTRONOMICAL IMAGING VIS-À-VIS SERENDIPITOUS ASTEROIDS, AND THEIR ASTROMETRIC PROPERTIES

by

JEAN-MARC DENIS
M.S. Ecole Superieure d'Ingenieur en Genie Electrique, 1987

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in the Department of Physics
in the College of Sciences
at the University of Central Florida
Orlando, Florida

Spring Term
2012

Major Professor: Yanga Fernández

ABSTRACT

The identification of known asteroids on existing CCD pictures would allow us to obtain accurate astrometric and photometric asteroid properties. Some asteroids might have ambiguous orbital elements, thus their identification along with their exact positions on multiple picture frames could significantly improve their orbital elements. Furthermore, the possibility of identifying known asteroids on older pictures, sometimes preceding their discovery date, might allow the study of non-gravitational effects like the Yarkovsky effect.

Identifying a potential Yarkovsky effect on asteroids is challenging because it is extremely weak. However, this effect cumulates with time, therefore, it is necessary to find astronomical pictures that are as old as possible. In addition, we need to collect high quality CCD pictures and use a methodology that would allow obtaining a statistically significant sample of asteroids.

To accomplish this, we decided to use the online archive of the Subaru telescope at Mauna Kea Hawaii because it has a prime-focus camera with a very high resolution of 80 millions pixels very well suited to capture serendipitous asteroids. In addition, the Subaru online archive has pictures from the last 10 years.

The methodology used in this thesis is to build a database that contains the orbital elements of all the known asteroids, allowing us to write a program that calculates the approximate position of all the asteroids at the date and time of each CCD picture we collect. To obtain a more precise position, the program also interfaces the JPL NASA Horizons on-line computation service. Every time an asteroid is found on a picture, Horizons sends its theoretical location back to the program. A later visual identification of this asteroid at this theoretical location on the picture triggers its input into our sample for further study.

This method allowed us to visually confirm 508 distinct asteroids on 692 frames with an average diameter of 3.6 km. Finally, we use the theory (given in appendix A) to calculate the theoretical drift of these asteroids that we compare with the one we measured on the CCD pictures.

This thesis is dedicated to my wife Rosine and my children Krystelle and Damien, whose love, support and interest helped me greatly to complete this research.

ACKNOWLEDGMENTS

I would like to express my deep appreciation and sincere thanks to my adviser Dr. Yanga Fernández. His guidance, availability and valuable suggestions helped me so much through this research.

I want to thank my wife Rosine and my niece Camille who helped me with the proof reading of this manuscript.

I also would like to acknowledge Ted Bowell, Bruce Koen, Dustin Lang and the JPL Solar System dynamics Group for their great help in providing the very useful information available on their website.

Finally, I want to thank my supervisor Jorge Collado for giving me the possibility to accommodate my working schedule with my UCF classes, and Orange County Utilities for their financial support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

NEO	Near-Earth objects
CCD	Charge-Coupled Device
FITS	Flexible Image Transport System
NASA	National Aerospace Administration
RA	Right Ascension
DEC	Declination
SQL	Structured Query Language
RDBMS	Relational Database Management system
PHP	Hypertext Preprocessor or Personal Home Page Tools
ASCII	American Standard Code for Information Interchange
FTP	File Transfer Protocol
GB	Gigabyte
WISE	Wide-Field Infrared Survey Explorer
AU	Astronomical Unit
Myr	Million of Year
JPL	Jet Propulsion Laboratory
DVD	Digital Video Disc

CHAPTER 1- INTRODUCTION

Modeling the orbital dynamics of asteroids reveals many complex problems. Besides the gravitational interaction with other planets, these motions are also affected by non-gravitational effects that are very subtle but can potentially affect the orbital dynamics of some asteroids over a long period of time. The purpose of this thesis is to study the Yarkovsky effect, a thermal radiation force, which can create a small thrust on some asteroids. Over a long period of time, this thrust can push an asteroid toward or away from the Sun causing a displacement or "drift" not included in the theoretical predicted orbital dynamics models of asteroids. In particular, this thermal radiation force increases the difficulty of modeling the motion of near-Earth objects (NEOs).

There are some properties we do not fully know about many asteroids such as their rotations, masses, thermal properties and the depth of their regolith. Since the theory predicts that the Yarkovsky effect depends on many of these parameters, every time we can detect and measure the drift caused by the Yarkovsky effect, we can then potentially deduct or make a hypothesis on some of these unknown properties.

As examples, asteroids such as Golevka, Apollo and 1950 DA have been studied by radar for years and their drifts have been measured. In addition, the determination of their spin state and diameter gave information regarding their bulk density, internal structure and thermal properties that could be derived from the strength of the Yarkovsky effect predicted in the theory.

Besides, radar that can only be used on a limited number of asteroids, a way to detect the weak potential Yarkovsky effect is to find a methodology that would allow the identification of asteroids on astronomical pictures old enough so that a larger drift caused by this effect over a long period of time can be observed. In addition, any methodology used would also need to produce a statistically significant sample of identifiable asteroids which would increase the chance of producing a potentially measurable effect.

In the following chapter (Materials and Methods), we will briefly discuss the theory behind the Yarkovsky effect. (a more complete mathematical description is given in appendix A). In the same chapter, we will also explain the method used to collect archive CCD astronomical images. In addition, we will describe the design of the system that includes programs, database and various interfaces that had to be designed to generate a list containing a significant sample of identifiable asteroids. (See appendix E: List of identified asteroids).

Subsequently, in the next chapter 3 (Results), we will present the detailed lists of the 508 asteroids that have been visually identified and confirmed on the collected CDD pictures along with their measured offsets. We will also use some statistical analysis and comparisons with the mathematical theory to confirm or disprove if any Yarkovsky effects have been found in these results. Finally, we will present in the conclusion recommendations and improvements for further research.

CHAPTER 2- MATERIALS AND METHODS

2.1- Definitions

The Subaru telescope, located in Mauna Kea in Hawaii, has a prime-focus CCD camera with a very high resolution of 80 million pixels very well suited for this kind of research [1]. In addition, it collects high quality pictures and has observational data archive from the last 10 years [2].

The Lowell Observatory is an astronomical observatory founded by Percival Lowell in 1894 in Flagstaff in Arizona. The Lowell Observatory has a total of 10 telescopes including one under construction. There are 5 telescopes on the Mars Hill site, including the observatory's original 24-inch Alvan Clark Telescope, and the 13-inch Pluto Discovery Telescope. 4 research telescopes are located on the Anderson Mesa dark sky site southeast of Flagstaff, including the 72-inch Perkins Telescope and the 42-inch John S. Hall Telescope. The observatory is also building a 4.3 meters (170 in) aperture telescope, which will be the fifth largest telescope in the continental United States. The asteroid observing services of the Lowell Observatory website contains an asteroid orbital elements database of all known asteroids whose orbital parameters are updated daily [3].

FITS (Flexible Image Transport System) is a standardized digital file format widely used in astronomy and adopted by NASA. Unlike other image formats, such as JPG or GIF, FITS is mainly designed to store, transmit, and manipulate scientific data consisting of multi-dimensional measurements (1-D spectra, 2-D images or 3-D data cubes) [4].

The JPL Horizons On-Line Ephemeris System is an online computation service system, developed conjointly by the Jet Propulsion Laboratory and NASA, which provides access to ephemerides for solar system objects including planets and asteroids. From these ephemerides, it is possible to calculate the position of an object in a spherical polar coordinate system of right ascension (RA) and declination (DEC). This Horizons System can be programmatically interfaced with telnet (horizons.jpl.nasa.gov:6775), email or manually through its web interface [5].

DS9 or (SAOImage DS9) is a standalone software used for astronomical imaging and data visualization. The application supports FITS format files and runs under Windows and Linux platforms. DS9 offers advanced features options such as multiple frame visualizations, blinking, mosaic images, color maps, zooming and a variety of coordinate systems [6].

Microsoft Access is a Windows database program with a query interface that can connect to a MySQL database. *Access Queries* offer the ability to retrieve and filter data from databases [7].

PHP is a programmatic language used primarily in web development to create dynamic interactions with databases. PHP can be embedded in HTML pages using special tags. Through the integration of HTML and PHP in websites, users can submit queries to a database; the data is interactively sent back and displayed on the html pages. PHP also has a command-line interface that can be used in standalone graphical applications [8].

MySQL is a relational database management system (RDBMS), also an open source database whose source code is available under the terms of the GNU General Public License. MySQL is used in many World Wide Web projects and products, including Wikipedia, Facebook, and Twitter. The database can be downloaded and installed on a local computer or used directly though the web. RDBMS databases store data and relationships among the data in structured tables that avoid data duplications. The data can be accessed or reassembled in many different ways using SQL (Structured Query Language). SQL is a programming language based upon relational algebra. SQL queries can be built and integrated within a program to retrieve or insert data in the RDBMS database using specific criteria [9].

Stored procedures and *stored functions* are programs that are made for relational database systems. A stored procedure, for example, is directly stored in the database itself. This allows a much faster execution time because there is no back and forth movement of data between the database and an external system. Instead, the stored procedures and the stored functions are running where the database data resides and are integrated with SQL statements. In addition, stored procedures can be nested, by executing one stored procedure from within another.

A *CCD* (charge-coupled device) picture is a numeric representation (normally binary) of a two-dimensional image obtained from a digital camera. Invented by Willard Boyle and George E. Smith, for which they received the Nobel Prize in 2009, CCD pictures have revolutionized astronomy. A CCD is an electronic device composed of millions of solid state pixels integrated in a single microchip. The sensitivity of CCD's is more than fifty times that of photographic films. This technology is allowing the Subaru prime-focus camera to produce large field of view gigapixel pictures of the sky, which is ideal to identify serendipitous asteroids. A CCD picture in prime-focus camera is a mosaic of 10 individual CCD frames made of 80-mega pixels (10,240 x 8,192). The camera covers a field of view 34' x 27', comparable to the angular size of the Moon and has a resolution of 0."202 per pixel [33]. For an object in the main belt (2 to 3 AU), this resolution corresponds to ~ 250 km (See also figure 1).

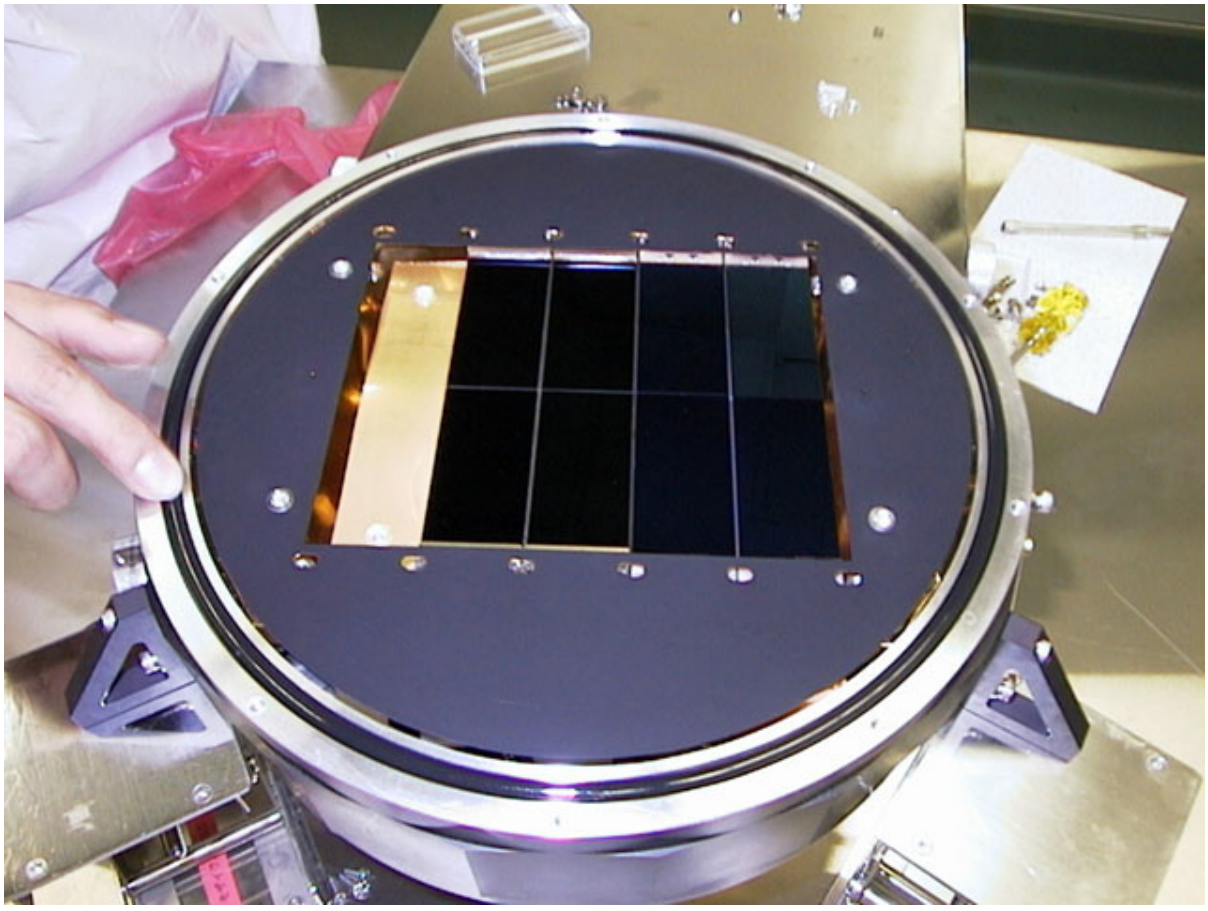


Figure 1: Prime-focus chamber filled with 8 of the 10 CCD detectors

(http://www.naoj.org/Topics/2000/07/15/SCam2.jpg)

2.2- Motivation

This high-resolution astronomical Subaru imaging data source, the JPL NASA Horizons on-line computation service, and the opportunity to integrate this data source with an online database system were the scientific motivation to start this project and identify asteroids in the data archive. The idea was to design a set of PHP programs and a MySQL database system to calculate the position of all known asteroids at any given date and time. The asteroid orbital elements database data of all the known asteroids were uploaded in the MySQL database from the Lowell Observatory. (See below, figure 2 with a brief schematic of the operation).

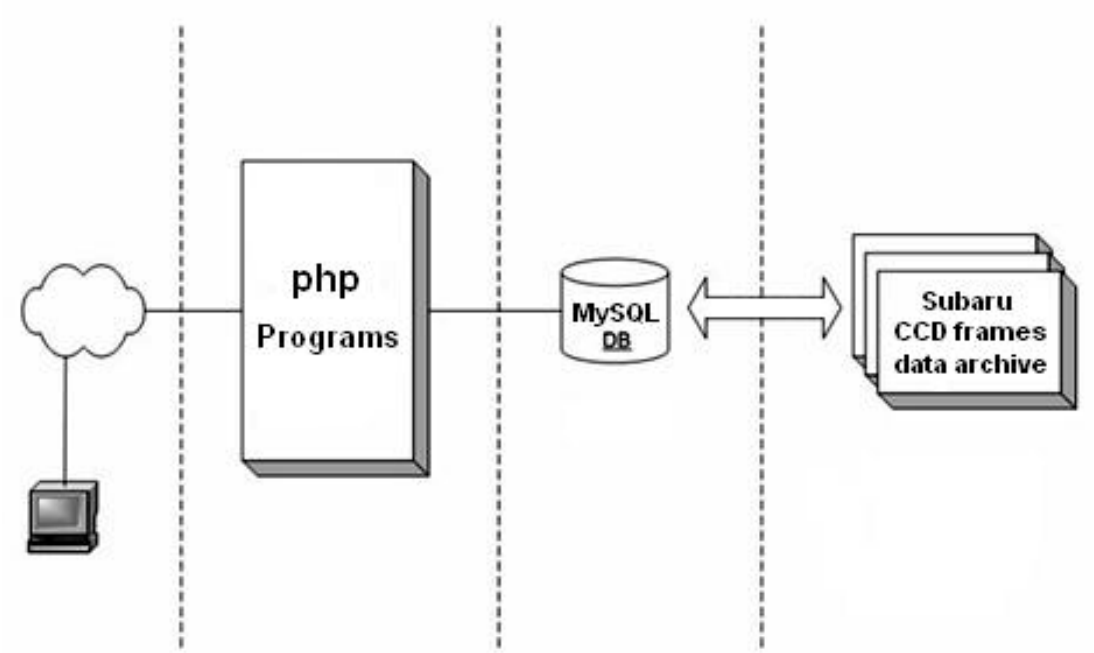


Figure 2: MySQL database integration

2.3- The Yarkovsky effect

The Yarkovsky effect is a non gravitational force that is created by the emission of thermal photons. The effect is a result of the delay created by the time it takes for the surface of the body to cool down.

The effect was discovered in 1901 by a Polish engineer, Ivan Osipovich Yarkovsky. He discovered that a subtle thermal effect would act on small bodies orbiting around the Sun. The radiation of the Sun acting on a rotating body would create a tiny non gravitational force that could lead to large long-term effects in the orbits of small asteroids (smaller then 40 km in diameter). The Yarkovsky effect has two components: The seasonal effect (See figure 3) and diurnal effect (See figure 4). The effect has been observed on at least two asteroids Golevka (6489) [17] and 1992 BF (152563) [18].

The diurnal effect is maximal when the asteroid's spin axis is perpendicular to the orbital plane (figure 4). The seasonal effect is maximal when the asteroid's spin axis is in the orbital plane (figure 3). Since the orbit of an asteroid rarely has its spin axis exactly perpendicular to the orbital plane or will rarely be exactly in the orbital plane, the Yarkovsky effect will act on small asteroids as a combination or superposition of both components, diurnal and seasonal.

2.3.1- The Seasonal component of the Yarkovsky Effect

The direction of the seasonal component of the Yarkovsky effect is always in the direction of the spin axis. The force is maximal at midnight and always pushes the asteroid toward the Sun.

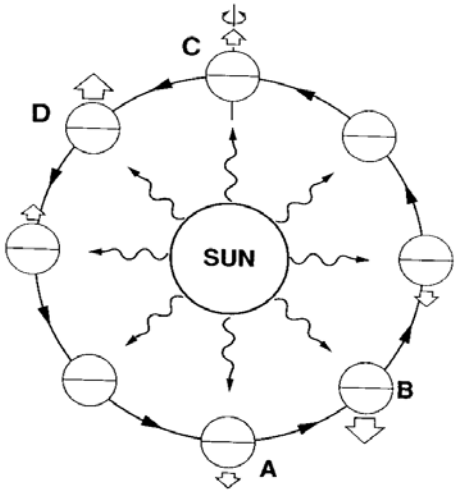


Figure 3: The seasonal Yarkovsky effect

The arrows show the direction of the thrust

(http://denali.gsfc.nasa.gov/annual2001/ar01text.htm)

The intensity variation of the incoming solar radiation depends on the mean motion of the asteroid around the Sun. This implies that the seasonal Yarkovsky component is related to its mean motion n (in degree per day) around its orbit. If the axial tilt of the asteroid (obliquity) is 0^o or 180^o, the seasonal component is zero. In this particular case, if we ignore the asteroid rotating (diurnal effect), no thermal radiation is generated by the seasonal component because there is symmetry between the northern and the southern hemispheres (there are no seasons, thus no thermal delay).

2.3.2- The Diurnal component of the Yarkovsky Effect

The direction of the thermal force of the diurnal component of the Yarkovsky effect depends on the sense of the rotation of the asteroid and the sense of direction of its orbit around the Sun. (See Figure 4).

If both senses are the same (e.g. both prograde rotation), the direction of the net radiation causes a drift spiraling away from the Sun causing the thermal force to increase the semi-major axis of the asteroid over a long period of time.
If both senses are opposite, (retrograde rotation), the thermal force pushes the asteroid toward the Sun causing a decrease of the semi-major axis of the asteroid over a long period of time.

If the axial tilt of the asteroid (obliquity) is 90^{o} or 270^{o}, the diurnal component is zero because the asteroid always shows the same hemisphere to the Sun over the course of its day. (thus, there is no thermal delay for this component).

Also, the intensity variation of the incoming solar radiation depends on the angular frequency ω of the asteroid around its spin axis. So the intensity of the diurnal Yarkovsky component is related to the angular frequency ω.

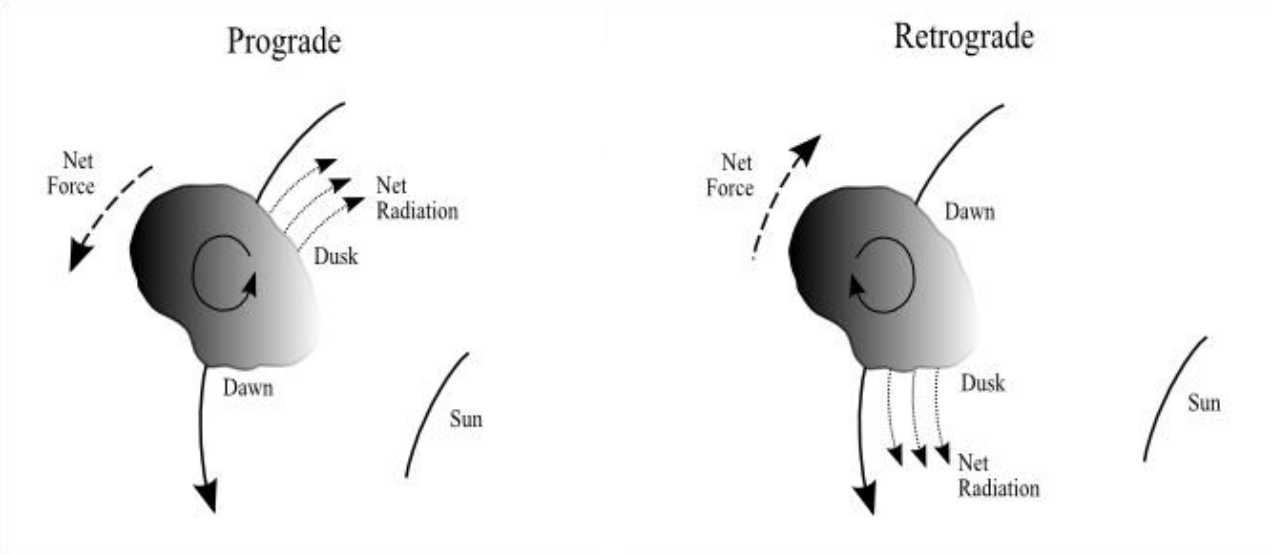


Figure 4: The diurnal Yarkovsky effect

(http://aeweb.tamu.edu/aemp/index.php?page=albedo)

2.4- Summary of procedure

The method used to identify asteroids on each CCD picture from the Subaru online archive is summarized in the following 5 steps:

Step 1: Obtain a list of all asteroid orbital elements and import this list in a MySQL table or list A. See: "2.5- The asteroid orbital elements table".

Step 2: Obtain a list of all frames in the online archive, with their center RA and Dec along with their date and time information. Cut out the pictures that are likely not to contain any asteroids, import this list in a MySQL table, or list B, along with the calculated RA and Dec of the edges of each picture in decimal degrees. See: "2.6- The Subaru image archive table".

Step 3: For a given time stamp of a frame, calculate the approximate RA and Dec of each asteroid in the list A for that date and time. Insert each potential match in a MySQL table, or list C, and repeat the same calculations for each frame on list B. See: "2.7- The asteroid matches table".

Step 4: For each potential asteroid match in list C, send an email to the JPL/NASA Horizons system. From the Horizons email reply, import the more accurate Horizons RA and Dec of this asteroid and update list C. See: "2.8- Email interface with the JPL/NASA Horizons system".

Step 5: For each potential asteroid match in list C, compare the Horizons RA and Dec with the RA and Dec of the edges of each picture. Insert each Horizons match in an Excel spreadsheet for future visual confirmation or list D. See: "2.9- Selecting asteroids only found within a picture".

We now describe these 5 steps in more details in the sections 2.5 to 2.9 below.

2.5- The asteroid orbital elements table

An ASCII file containing the asteroid orbital elements of all the known asteroids was downloaded from the Lowell Observatory website. The file is available at [3] and contained 478,009 asteroids at the date and time of the download. This file was imported to the MySQL table tbl_astorb with the following columns:

(f1) Asteroid number (blank if unnumbered).

(f2) Name or preliminary designation.

(f4) Absolute magnitude H, magnitude

(f10) Orbital arc, days, spanned by observations used in orbit computation.

(f11) Number of observations used in orbit computation.

(f12) Epoch of osculation, yyyymmdd (TDT).

(f13) Mean anomaly, deg.

(f14) Argument of perihelion, deg (J2000.0).

(f15) Longitude of ascending node, deg (J2000.0).

(f16) Inclination, deg (J2000.0).

(f17) Eccentricity and (f18) Semimajor axis, AU.

We then used the following script in the MYSQL Browser to build this table:

```
CREATE TABLE  'PHYSICS'.'tbl_astorb' (
  `f1` int(11) default NULL,
  `f2` varchar(50) default NULL,
  `f4` float default NULL,
  `f10` int(11) default NULL,
  `f11` int(11) default NULL,
  `f12` date default NULL,
  `f13` float(9,6) default NULL,
  `f14` float(9,6) default NULL,
  `f15` float(9,6) default NULL,
  `f16` float(9,6) default NULL,
  `f17` float(10,8) default NULL,
  `f18` double(25,10) default NULL));
```

An Access query was built to populate the tbl_astorb from the asteroid orbital elements ASCII file. The first 3 rows of this table are given below. (See table 1).

Table 1 MySQL table "tbl_astorb" with 3 ast. and orbital elements

F1	f2	f4	F11	F12	F13	f14	f15	f17	f18
1	Ceres	3.34	6100	01/04/10	70.43399	72.6927	80.39384	0.07922201	2.7657232
2	Pallas	4.13	7281	01/04/10	53.3954	310.207	173.1284	0.2309704	2.7727488
3	Juno	5.33	6317	01/04/10	346.9285	248.086	169.9116	0.2549657	2.6700005

f1, f2, f3, f4, f11, f12, f13, f14, f15, f16, f17 and f8 are the columns of table "tbl_astorb".

2.6- The Subaru image archive table

There are 44,562 CCD pictures in the Subaru archive files from 2004 to 2009. From the Subaru website [2], using the "List of All Frames" option, it is possible to generate and download a list of all the images the telescope took with its prime-focus camera for a given year. Using this option, six text files from years 2004 to 2009 were generated and downloaded.

These ASCII files were then imported to an Access table called "SUP_YEARS". The table has the following columns:

YEAR,	Year the picture was taken
#FRAME_ID,	Frame ID (1 to 10)
DATE_OBS,	Date the picture was taken
FILTER,	Type of filter used
RA2000,	RA coordinate
DEC2000,	DEC coordinate
UT_STR,	UT TIME
EXPTIME,	Exposure time (in seconds)
DATA_TYP,	Type of picture
OBJECT	Name of the object taken by the picture

Following this import, the SUP_YEARS table contained 44,562 picture rows. After a preliminary analysis of this data, it has been decided to filter out the data

of this table to only keep the picture rows that have the greatest chance to contain identifiable asteroids. Thus, the following criteria were used:

1- Pictures whose ecliptic latitude is within ± 30^{o} from the ecliptic.

2- Pictures that are "non-calibration frame“.

3- Pictures that have an exposure time of at least 100 seconds.

4- Pictures that have the most sensitive filters, red and infrared (~0.7 μm).

R filter: 5700Å ≤ R ≤ 7500Å

I filter: 7000Å ≤ I ≤ 9000Å

(http://www.naoj.org/Observing/Instruments/SCam/jpg/johnson.gif)

In order to implement this criterion, we built an Access queries with the following "where clause" conditions:

1- Equatorial (λ,β) coordinate of Pictures: $0 \leq \lambda \leq 360^{\circ}$ and $-30^{\circ} \leq \beta \leq +30^{\circ}$.

2- Pictures that are "non-calibration frame": DATA_TYPE= 'OBJECT'.

3- Picture exposure time: EXPTIME ≥ 100.

4- Pictures that have: (FILTER = 'W-C-RC' or FILTER = 'W-C-IC').

The first Access query called "QRY_SUP_YEARS" has been created to generate and populate a second table called "TBL_FRAMES_GEN". As it populates the table "TBL_FRAMES_GEN", the query also calculates the equatorial coordinates (λ,β) of each picture row.

A second query called "QRY_SUP_YEARS_FILTER" select the 2,078 pictures that follow the "where clause" criteria defined above. (See appendix B: Access queries for a complete detail description of these two queries).

A picture table is created in the MYSQL Query Browser and populated with the QRY_SUP_YEARS_FILTER query.

The script to build the 'tbl_pictures" table is:

```
CREATE TABLE  `tbl_pictures` (
 `p1` int(11),                          Unique identifier
 `FRAME_ID` varchar(50),                Frame ID
 `DATE_OBS` date,                       Date when picture taken
 `FILTER` varchar(50),                  Filter type
 `RA2000` varchar(50),                  Picture center RA coordinate
 `DEC2000` varchar(50),                 Picture center DEC coordinate
 `UT_STR` varchar(50),                  Time when picture was taken
 `EXPTIME` varchar(50),                 Exposition time
 `DATA_TYP` varchar(50),                Type of picture
 `OBJECT` varchar(50),                  Name of the picture object
 `RA_CALC` double(15,12),               RA coordinate in decimal
 `DEC_CALC` double(15,12),              DEC coordinate in decimal
 `RA_MIN_CALC` double(15,12),           RA picture bottom coord. in decimal.
```

```
`RA_MAX_CALC` double(15,12),        RA picture top coordinate in decimal.
`DEC_MIN_CALC` double(15,12),       DEC picture bottom coord. in decimal.
`DEC_MAX_CALC` double(15,12),       DEC picture top coordinate in decimal.
UNIQUE KEY `pk_FRAME_ID` (`FRAME_ID`), UNIQUE KEY `pk_p1` (`p1`));
```

The MySQL store procedure "pic_RA_CALC" is manually run, loops through the tbl_pictures table, fetches the RA value of each picture given by the Subaru website as a format "HH:MM:SS.SSS", calculates its decimal value and updates the column RA_CALC with this calculated value.

Similarly, the MySQL store procedure " pic_DEC_CALC", loops through the tbl_pictures table, fetches the DEC value of each picture given by the Subaru website as a format "±HH:MM:SS.SS", calculates its decimal value and updates the column DEC_CALC with this calculated value. (See appendix C: Stored procedures and programs).

Finally, the Access query "qry_populate_picture_local_access" updates the RA_MIN_CALC and RA_MAX_CALC of each picture row in the tbl_pictures table by adding or subtracting half the height of the picture (±0.355).
The same Access query "qry_populate_picture_local_access" also updates the DEC_MIN_CALC and DEC_MAX_CALC by adding or subtracting half the width

of the picture (±0.2319 degrees). The result, for the first three rows of table tbl_pictures, is given below on table 2 as an example.

Table 2 The MySQL "tbl_pictures" table with 3 sample pictures

p1	FRAME_ID	DATE_OBS	FILTER
10	SUPA0059157X	09-Jan-08	W-C-RC
11	SUPA0026688X	17-Jan-04	W-C-IC
12	SUPA0027768X	17-Feb-04	W-C-RC

RA2000	DEC2000	UT_STR	EXPTIME (s)
09:58:23.069	-11:03:16.70	12:45:27.298	300
04:39:39.152	+07:27:24.65	07:54:00.233	180
09:00:26.975	+39:10:51.22	06:45:35.167	300

DATA_TYP	OBJECT	RA_CALC (deg)	DEC_CALC (deg)
OBJECT	A907	149.5961	-11.0546
OBJECT	GRB031220	69.9131	7.4568
OBJECT	IRAS08572+3915	135.1124	39.1809

RA_MIN_CALC (deg)	RA_MAX_CALC (deg)	DEC_MIN_CALC (deg)	DEC_MAX_CALC (deg)
149.2917	149.9	-11.2831	-10.8219
69.6125	70.2167	7.2636	7.6542
134.7333	135.5	38.9878	39.3772

2.7- The asteroid matches table

A MySQL stored procedure "pic_FIND_MATCH" was written (See appendix C: Stored procedures and programs for more details). This procedure populates the table "tbl_matches" when an asteroid is found in a picture. The script to build the "tbl_matches" table is:

```
CREATE TABLE  `tbl_matches` (
  `PIC_frame_pk` int(11),                Picture ID unique identifier
  `PIC_FRAME_ID` varchar(50),            Subaru Picture ID
  `PIC_RA_CALC` double(15,12),           Pic. center RA coordinate in decimal
  `PIC_DEC_CALC` double(15,12),          Pic. center DEC coordinate in decimal
  `PIC_DATE_OBS` date,                   Date the picture has been taken
  `PIC_UT_STR` varchar(50),              Time the picture has been taken
  `AST_id` int(11),                      Asteroid ID unique identifier
  `AST_name` varchar(50),                Asteroid name
  `AST_RA_CALC` double(15,12),           Asteroid RA coordinate in decimal
  `AST_DEC_CALC` double(15,12),          Asteroid DEC coordinate in decimal
  `DELTA_RA` double(15,12),              Diff.: RA pic. center - RA asteroid.
  `DELTA_REC` double(15,12),             Diff.: DEC pic. center - DEC asteroid.
  `MAGNITUDE` float,                     Asteroid absolute magnitude
  `DELTA_PROB` float(5,2),               Probability to find the ast. on the pic.
  `HORIZONS_LINE` varchar(200),          Horizons email data
  `HORIZONS_MAG` double(15,12),          Horizons absolute magnitude
```

```
`HORIZONS_RA_val` double(15,12),    Horizons RA coordinate in decimal
`HORIZONS_DEC_val` double(15,12), Horizons DEC coordinate in decimal
`HORIZONS_RA` varchar(50),          Horizons RA coordinate
`HORIZONS_DEC` varchar(50),         Horizons DEC coordinate
`HORIZ_DELTA_RA` double(15,12),     Diff.: Horizons RA ast. - RA asteroid
`HORIZ_DELTA_DEC` double(15,12),  Diff.: Horizons DEC ast. - DEC asteroid
`PIC_DELTA_RA` double(15,12),       Diff.: RA pic. center - Horizons RA ast.
`PIC_DELTA_DEC` double(15,12)));    Diff.: DEC pic. center - Horiz. DEC ast.
```

2.8- Email interface with the JPL/NASA Horizons system

A program called "aster_coordinate.php" was written in php to interface the JPL/NASA Horizons system [5] (See appendix D: Php programs). This php program was published online and is available from a website to better interface with the Horizons system and the Mysql database. For any asteroid at a given date and time, the php program automatically builds and sends an email request to Horizons. Each email reply from Horizons is received and stored in a stand alone Outlook system. The Outlook system has an export option allowing the generation of a unique text file that contains all the Horizons email replies. Finally the "aster_coordinate.php" program can also automatically import this Outlook export file back to the MySQL database.

2.9- Selecting asteroids only found within a picture

The final Access query "qry_asteroids_to_keep" filters the tbl_matches table by comparing the decimal values of the RA and DEC calculated by the Horizons online system with the min, max of RA and the min and max of DEC which represent the external boundaries of the Subaru CCD picture. Any asteroid whose calculated Horizons RA and DEC are found within the boundaries of the pictures are selected and saved in an Excel spreadsheet for future visual confirmation. (See annexe D for more detail and results in table 3).

Table 3 The MySQL "tbl_matches" table with 3 sample rows

PIC_frame_pk	PIC_FRAME_ID	PIC_RA_CALC (deg)	PIC_DEC_CALC (deg)
594	SUPA0032920X	328.4186	17.6775
594	SUPA0032920X	328.4186	17.6775
594	SUPA0032920X	328.4186	17.6775
594	SUPA0032920X	328.4186	17.6775

PIC_DATE_OBS	PIC_UT_STR	AST_id	AST_name
18-Jul-04	11:38:39.722	194123	2001 SA263
18-Jul-04	11:38:39.722	232255	1997 RG8
18-Jul-04	11:38:39.722	341881	2005 WO12
18-Jul-04	11:38:39.722	411993	2007 UM126

AST_RA_CALC (deg)	AST_DEC_CALC (deg)	DELTA_RA (deg)	DELTA_REC (deg)
328.1475404	17.86781679	0.27105959	0.190316788
328.0793653	17.82672513	0.339234654	0.14922513
329.8739563	17.82617029	1.45535633	0.148670286
327.0676349	17.35527812	1.350965107	0.322221877

MAGNITUDE	DELTA_PROB	Match_id	HORIZONS_RA_val (deg)
19.8075	66.96	1	328.834
17.9469	63.05	2	329.1868
20.497	45.76	3	330.1381
19.4386	38.4	4	327.241

HORIZONS_DEC_val (deg)	HORIZONS_RA	HORIZONS_DEC	HORIZ_DELTA_RA (deg)
17.89141	21 55 20.1718	+17 53 29.085	0.68645959
17.82572	21 56 44.8280	+17 49 32.598	1.107434654
17.76658	22 00 33.1356	+17 45 59.686	0.26414367
17.40359	21 48 57.8476	+17 24 12.906	0.173365107

HORIZ_DELTA_DEC (deg)	PIC_DELTA_RA (deg)	PIC_DELTA_DEC (deg)	HORIZONS_MAG
0.023593212	0.4154	0.21391	20.74
0.00100513	0.7682	0.14822	18.91
0.059590286	1.7195	0.08908	20.95
0.048311877	1.1776	0.27391	19.82

2.10- Architecture and algorithm

We have now built and populated two MySQL tables: tbl_astorb containing the 478,009 known asteroids and tbl_pictures containing 2,078 pictures that follow the criteria described in 2.5. The stored procedure "pic_FIND_MATCH" uses a double loop, following the flowchart given on the next page. The first loop fetches the picture number j (from 1 to 2,078), the observed date and time, the Subaru picture ID and the position of the center of the picture in the sky given by RA_CALC(i) and DEC_CALC(i).

The second loop fetches the asteroid number i (from 1 to 478,009) and the orbital elements needed to calculate the position of the asteroid in the sky at the observed date and time of the selected pictures. (See figure 5 below).
Within this second loop, from the fetched orbital elements, a series of stored functions calculates the following information (for each asteroid):
Ut_time_earth(), Ut_time() and Ut_time_epsilon()
f_geo_ecliptic_long(): Geocentric elliptic longitude.
f_geo_ecliptic_lat(): Geocentric elliptic latitude.
f_true_anomaly (): True anomaly.
f_heliocentric_dist (): Heliocentric distance.
f_heliocentric_lat (): Heliocentric latitude.
f_magnitude (): Magnitude.
ec_anomaly(): Anomaly

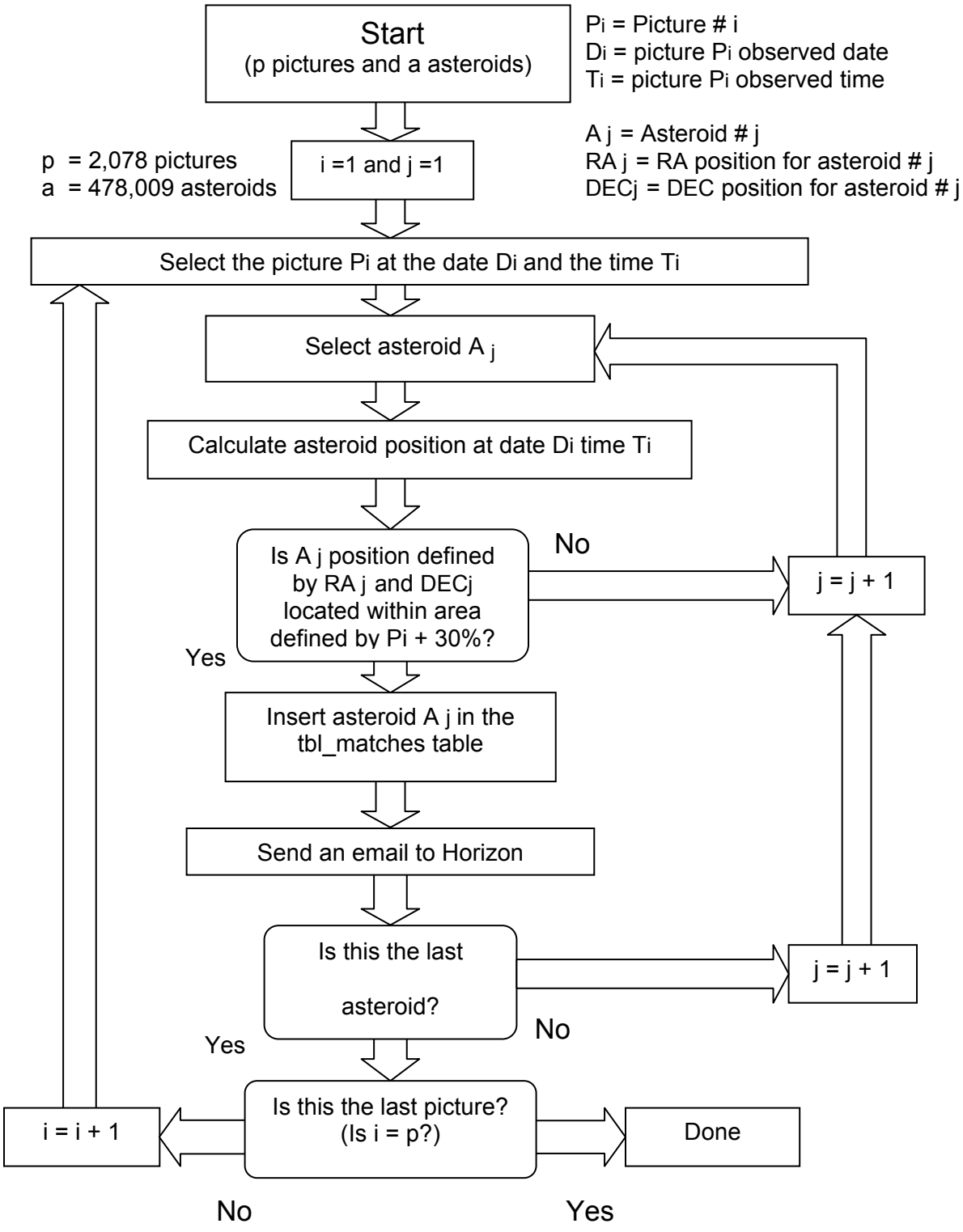


Figure 5: The picture and asteroid scan algorithm

The values calculated by these functions are used to compute the positions v_right_ascention_calc(j) and v_declination_calc(j) for each asteroid A(j) of this second loop. These positions are then compared to the positions RA_CALC(i) and DEC_CALC(i) of the selected picture Pi. Any asteroid A(j) whose positions are within one arc second of the center of the picture Pi is flagged and inserted in the table "tbl_matches" along with the asteroid's id, picture id, asteroid's coordinates, calculated magnitude, observed date and time and distance "delta_RA /delta_DEC" from the center of the picture Pi.

The php program "aster_coordinate.php" then loops though the table tbl_matches and automatically builds and sends an email request to the JPL/NASA Horizons system for each row of this table. The email replies from Horizons are later stored in an Outlook system. When this process is completed, an export file is generated from Outlook that contains all the Horizons' email replies. The export file is then uploaded to the online website and read by the aster_coordinate php program to update the "tbl_matches" table with the more precise Horizons RA and DEC coordinates. (See table 3).

Finally, the Access query " qry_asteroids_to_keep" filters the tbl_matches table and selects the asteroids whose Horizons RA and Horizons DEC are within the boundaries of their corresponding pictures. Each asteroid found within a picture is saved in an Excel spreadsheet for future visual confirmation.

CHAPTER 3- RESULTS

3.1- Picture results after applied criteria

When the Access query "QRY_SUP_YEARS_FILTER" is run against the SUP_YEARS table containing the 44,562 pictures that were downloaded from the Subaru website, we obtain the following set of pictures and objects for the years 2004-2009:

Table 4 The picture and object results

Year	Objects	Pictures	
2004	132	790	
2005	81	434	
2006	77	271	
2007	18	149	
2008	32	346	
2009	14	88	
Total:	121	2,078	(out of 44,562 pictures or 4.6%)

These 2,078 pictures have been inserted in the table tbl_pictures. Each picture representing the same area of space (same object) are selected twice at date Di and at time ti and ti + Δt to confirm the movement of a potential asteroid from one picture to the other (15 mn ≤ Δt ≤ 120 mn).

Note: Sometimes it was necessary to select the same picture at three different times because the asteroid could not be visually identified on the second picture but could be confirmed on a third picture.

For example: In 2004, the object "A25523" exists on 3 frames: SUPA0033087X, SUPA0033088X and SUPA0033089X. Each picture is composed of 10 frames. (See figure 1).

3.2- Emailed results after scanning the picture and asteroid tables

When the stored procedure "pic_FIND_MATCH" was run, 45,200 matches of potential asteroids were found on 262 pictures.

When the online php program "aster_coordinate.php" was run, 45,200 email requests were sent to the Horizons system. The email replies data was imported and the tbl_matches table was updated with this Horizons data.

When the qry_asteroids_to_keep query was run, 1,256 asteroids were confirmed to be located on 262 pictures or 262 x 10 = 2,620 frames.
These 1,256 asteroids along with their frame ID, Horizons positions RA and DEC and calculated magnitude were saved on an Excel spreadsheet file "db_frame.xls". The 2,620 frames were downloaded from the Subaru website [2]. (See figure 7).
Each downloaded CCD frame is a FITS file that has a size of 16,673 KB = 0.0167GB, so each picture made of 10 frames, has a size of 16.7MB. Thus, 2,620 frames x 0.0167GB = 44 GB of data have been downloaded.

3.3- Visual identification with DS9

Using the DS9 software [6], we can locate a potential asteroid on a frame Pi taken at date Di and time ti, located at Horizons positions RA(i) and DEC(i). When the asteroid is located, it needs to be confirmed on the frame taken at time ti + Δt. We then display simultaneously the two frames with DS9; we have the option to rapidly switch the view back and forth between these 2 frames, making it possible to detect the movement of the asteroid in relation to the background of the stars which seems motionless. Using this technique, 508 distinct asteroids have been identified on 692 frames. Every time the identification of an asteroid was done, a snapshot of both frames was saved on a Word document with the name of the frames and the identified asteroid. (See figure 6).

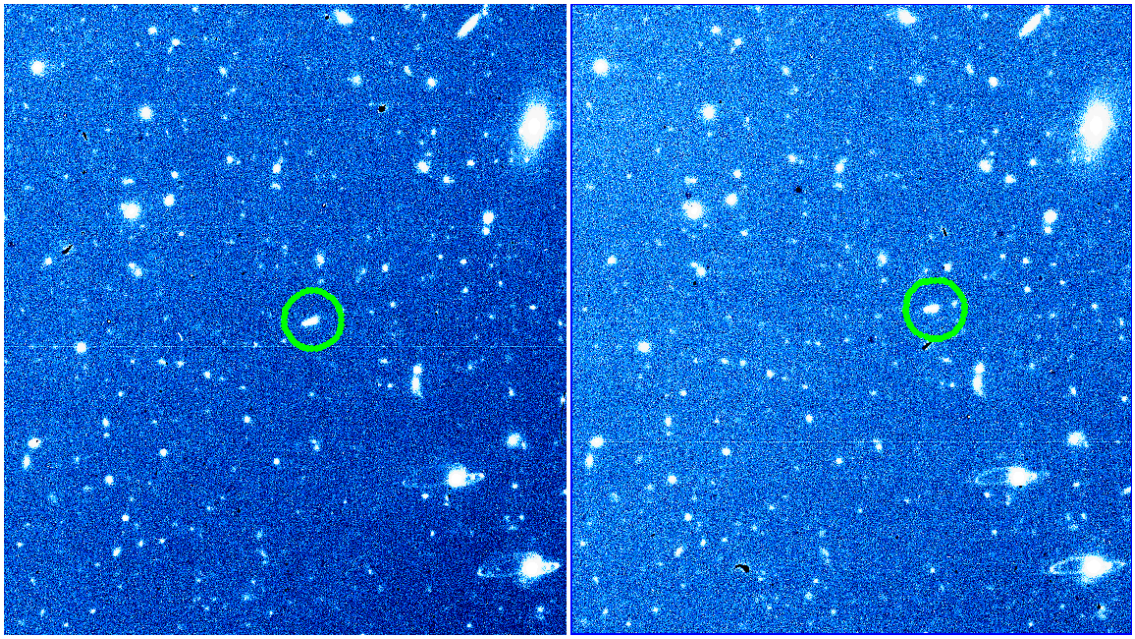


Figure 6: Asteroid 2002 TK134 on frames 278093 & 278133

3.4- Results and data processes

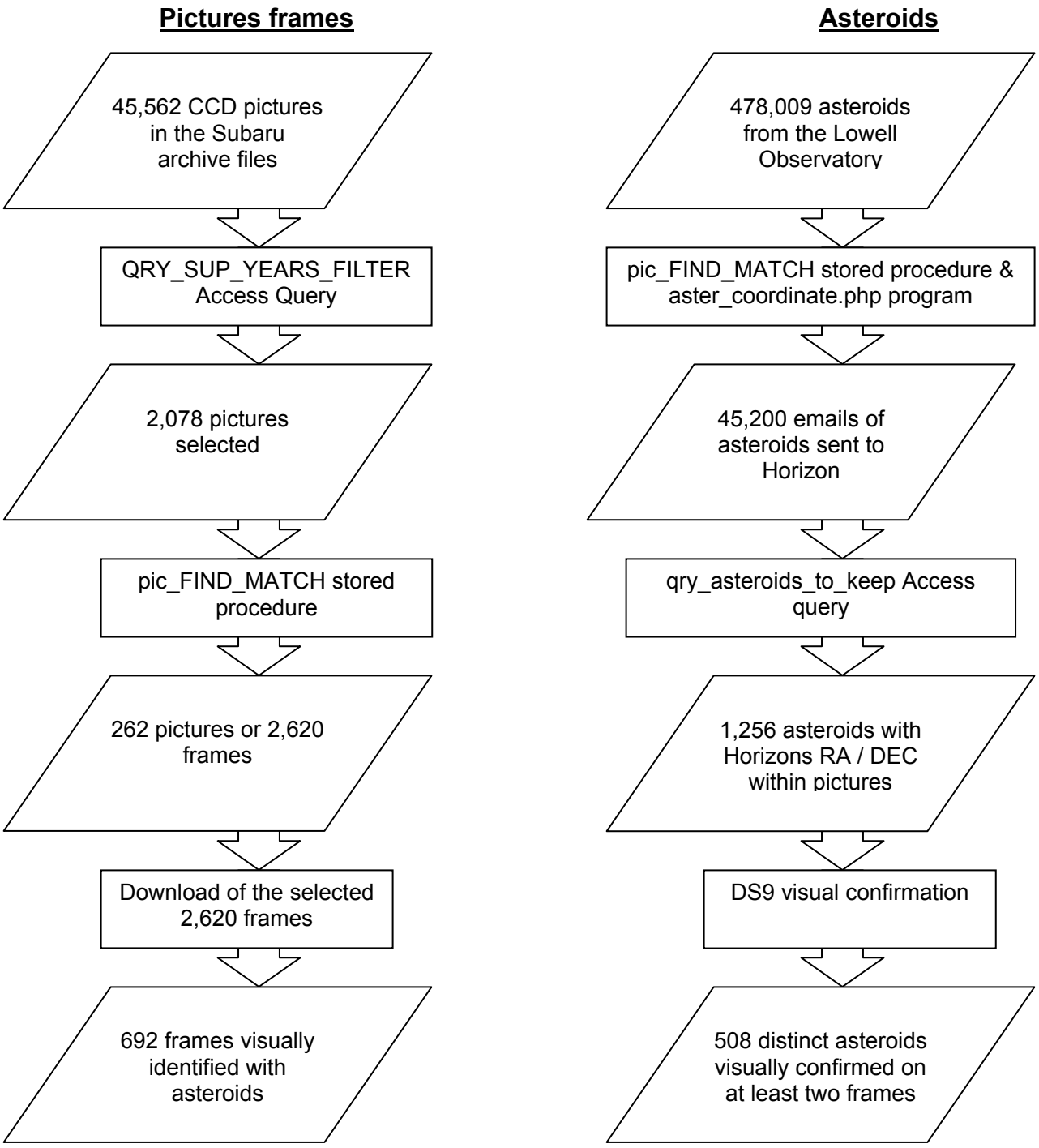


Figure 7: Picture frames and asteroids data processes

3.5- Identified asteroid results

The 508 snapshots of distinct asteroids have been saved as Word documents along with their corresponding FITS files (692 frames), representing 27GB of data. (See the completed list of these asteroids in appendix E: List of identified asteroids).

All these files have been saved on a set of 7 DVDs. 26 asteroids were found on multiple pairs of frames (duplicate matches). 96 asteroids have been identified on pictures that pre-date the year of their confirmed discovery.

One of the identified asteroids, 14827 Hypnos (1986 JK) is a Near Earth Orbit Object (N.E.O.) [10]. The others are in the main belt.
We obtained the following set of asteroids captured in pictures for the years 2004-2009:

Table 5 The number of identified asteroids per picture year

Pictures Year	Asteroids
2004	60
2005	325
2006	40
2007	19
2008	52
2009	12
Total:	508

3.6- Measure of the offsets results

The offsets are measured with DS9. For each identified asteroid, we take down its actual positions RA and DEC indicated by DS9 [6] on both frame at times ti and ti + Δt. The results are entered in an Excel file called "statistical_analysis.xls" pre-populated with the list of the 508 identified asteroids along with their observed date / time, frame ID, magnitude, and Horizons positions RA and DEC on each frame. The Yarkovsky effect is not considered in the calculated Horizons positions RA and DEC. From this data, Excel then automatically calculates the offset Θ in arc seconds for each entry. As an example, two asteroids taken out of the "statistical_analysis.xls" fspreadsheet are shown below. (See table 6).

Table 6 A sample of 2 asteroids with their measured offsets

Asteroid Name	P1 (Pic ID)	PIC_FRAME_ID	Pic Date	PIC_UT_STR
Epetersen	710	SUPA00365192	05-Jan-05	14:05:56.094
Epetersen	711	SUPA00365112	05-Jan-05	13:32:57.313
1997 WZ27	174	SUPA00418919	30-Aug-05	12:18:18.149
1997 WZ27	178	SUPA00418959	30-Aug-05	12:42:28.003

Asteroid Name	Horizons RA Mauna Kea	Horizons DEC Mauna Kea	Measured RA	Measured DEC
Epetersen	07 48 09.6908	+22 22 29.846	07 48 09.661	+22 22 29.73
Epetersen	07 48 11.2974	+22 22 27.230	07 48 11.273	+22 22 27.05
1997 WZ27	04 32 06.2517	+22 40 22.564	04 32 06.287	+22 40 22.60
1997 WZ27	04 32 07.0984	+22 40 25.264	04 32 07.132	+22 40 25.35

Asteroid Name	Horizons MAG.	Δ RA (arcsec)	Δ DEC (arcsec)	Dist. (arcsec)	Avg Offset (arcsec)
Epetersen	16.39	0.41	0.12	0.43	**0.41**
Epetersen	16.39	0.34	0.18	0.38	**0.41**
1997 WZ27	20.79	0.49	0.04	0.49	**0.48**
1997 WZ27	20.79	0.47	0.09	0.47	**0.48**

P1 (Pic ID): Frame unique identifier ID	Horizons MAG: Absolute Magnitude returned by Horizons
PIC_FRAME_ID: Subaru frame ID	Measured RA: Measured RA on frame using DS9
Pic Date: Date picture was taken	Measured DEC: Measured DEC on frame using DS9
PIC_UT_STR : UT time picture	ΔRA: Absolute Difference Horizon RA -Measured RA
Dist (arcsec): Dist. squared ΔRA & ΔDEC	Horizons DEC: Horizons DEC, center-site name: Mauna Kea
Avg Offset : Average dist. for both frames	Horizons RA: Horizons RA, center-site name: Mauna Kea

3.6.1- Offsets of duplicate asteroids and optical distortion

To validate these offsets, we considered some of the 26 duplicate asteroids. If the same asteroid is identified on multiple pictures then, as long as the picture's dates are not too far apart, its offset on each picture should be very similar. In other word, if we see the same asteroid in two separate pairs of images, then the asteroid's offset from the predicted Horizons position should be about the same in each pair, since it's unlikely for the asteroid to so drastically change the amount of offset from its predicted location in such a short time.

Therefore, we measured the offsets of asteroids 1998 DU20 and 1997 NS7. These 2 asteroids where identified on two pairs of frames taken 24 hours apart.

For 1998 DU20 we find an average offset of 13.92 arc sec. on December 1 and an average offset of 18.59 arc sec. on December 2.

For 1997 NS7 we find an average offset of 9.57 arc sec. on December 1 and an average offset of 8.30 arc sec. on December 2.

The detailed result is given below for asteroid 1998 DU20 (See table 7).

Table 7 1998 DU20 duplicate ast. with inconsistent measured offsets

Asteroid Name	P1 (Pic ID)	PIC_FRAME_ID	Pic Date	PIC_UT_STR
1998 DU20	225	SUPA00438129	01-Dec-05	13:33:10.005
1998 DU20	226	SUPA00438189	01-Dec-05	14:04:24.359
1998 DU20	596	SUPA00439345	02-Dec-05	13:34:13.808
1998 DU20	597	SUPA00439365	02-Dec-05	13:44:18.303

Asteroid Name	Horizons RA Mauna Kea	Horizons DEC Mauna Kea	Measured RA	Measured DEC
1998 DU20	11 19 17.6438	+03 30 46.618	11 19 17.523	+03 31 01.13
1998 DU20	11 19 18.7762	+03 30 39.749	11 19 18.656	+03 30 52.85
1998 DU20	11 20 10.3193	+03 25 31.635	11 20 10.802	+03 25 36.24
1998 DU20	11 20 10.6784	+03 25 29.459	11 20 11.153	+03 25 34.32

Asteroid Name	Horizons MAG.	Δ RA (arcsec)	Δ DEC (arcsec)	Dist. (arcsec)	Avg Offset (arcsec)
1998 DU20	16.39	1.81	14.51	14.62	**13.92**
1998 DU20	16.39	1.80	13.10	13.22	**13.92**
1998 DU20	20.79	7.23	4.60	8.57	**8.59**
1998 DU20	20.79	7.11	4.86	8.61	**8.59**

P1 (Pic ID): Frame unique identifier ID
PIC_FRAME_ID: Subaru frame ID
Pic Date: Date picture was taken
PIC_UT_STR : UT time picture was taken
Dist (arcsec): Dist. squared ΔRA & ΔDEC
Avg Offset : Average dist. for both frames
Horizons MAG: Absolute Magnitude returned by Horizons
Measured RA: Measured RA on frame using DS9
Measured DEC: Measured DEC on frame using DS9
ΔRA: Absolute Difference Horizon RA -Measured RA
Horizons DEC: Horizons DEC, center-site name: Mauna Kea
Horizons RA: Horizons RA, center-site name: Mauna Kea

These errors were confirmed with calibration software showing the positions of known stars that were off by 7 arc seconds on frame SUPA00439345.

The Offset errors are consistent with a field distortion of the optical system; they decrease as the asteroid is located closer to the center of the frame.

As an example, figure 8 gives a level of optical distortion over the field of view amounting to roughly 10 arc seconds (See scale indicated in green at the center of figure 8). This example is not from the prime-focus camera but the principle is the same. The distortion field in this case is closed to zero at the center of the picture and increase toward 10 arc seconds at the edge of the 0.8 degree diameter field. The circle in blue indicates the edge of the distortion field of the optical system. To solve this problem and to be able to make accurate offset measurements, we need to calibrate the Subaru picture frames.

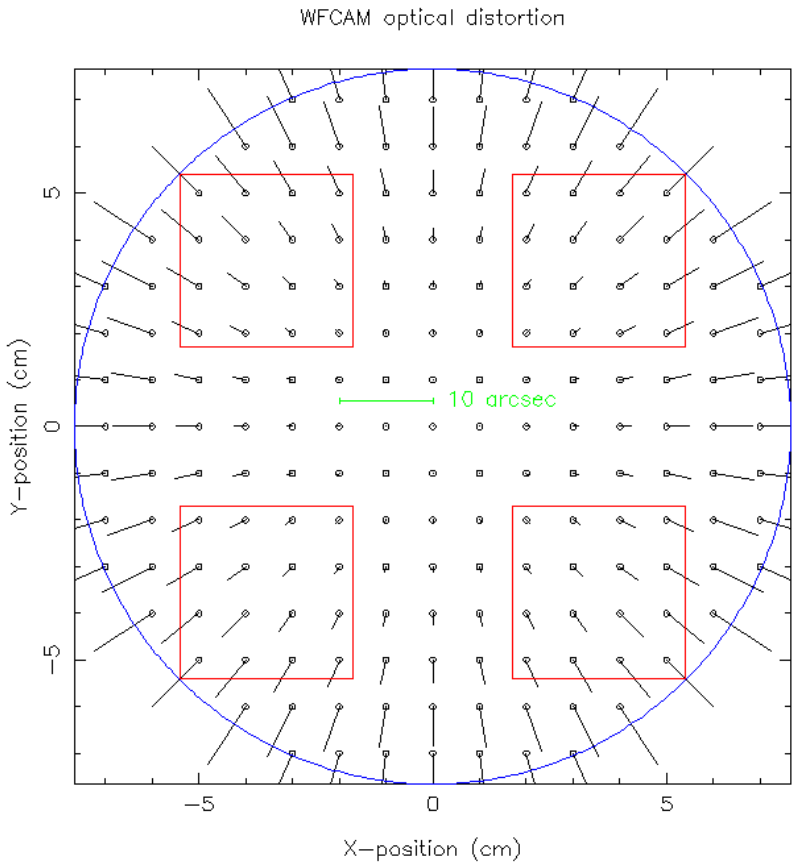


Figure 8: An example of radial distortion on a picture frame

(http://casu.ast.cam.ac.uk/surveys-projects/wfcam/technical/astrometry)

3.6.2- Astrometry.net online system

We used astrometry.net [11], an online system that processes and corrects submitted astronomical images. Patterns of four or five stars called asterisms are identified in the pictures and geometrically encrypted or hashed. These patterns are then compared and matched with a catalog of pre-indexed hashes. This process generates a new image free of distortion [34]. Below is an example of a frame submitted to astrometry.net, see figure 9. The image on the right has been corrected. Stars on the edges of the frames have been slightly displaced after image corrections.

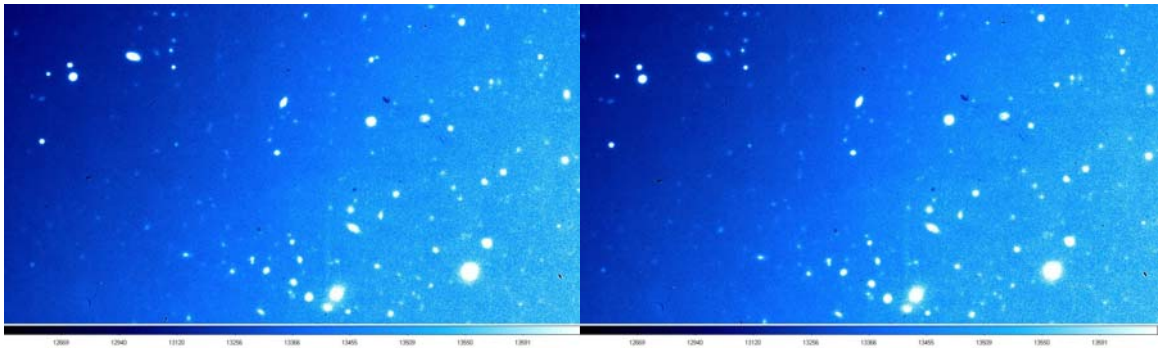


Figure 9: A frame before (left) and after distortion correction (right)

We used the astrometry website [11] to process the four frames that contained the two previous duplicate asteroids as a pre-test.

The detailed result is given below for both asteroids 1998 DU20 and 1997 NS7

The average offsets are now consistent. (See table 8 before and after frame distortion correction).

Table 8 Offsets of ast. 1998 DU20 &1997 NS7 before/after correction

Asteroid Name	P1 (Pic ID)	PIC_FRAME_ID	Pic Date	PIC_UT_STR
1998 DU20	225	SUPA00438129	01-Dec-05	13:33:10.005
1998 DU20	226	SUPA00438189	01-Dec-05	14:04:24.359
1998 DU20	596	SUPA00439345	02-Dec-05	13:34:13.808
1998 DU20	597	SUPA00439365	02-Dec-05	13:44:18.303
1997 NS7	225	SUPA00438124	01-Dec-05	13:33:10.005
1997 NS7	226	SUPA00438184	01-Dec-05	14:04:24.359
1997 NS7	596	SUPA00439361	02-Dec-05	13:44:18.303
1997 NS7	597	SUPA00439341	02-Dec-05	13:34:13.808

Asteroid Name	Horizons RA Mauna Kea	Horizons DEC Mauna Kea	Measured RA Before correction	Measured RA After correction
1998 DU20	11 19 17.6438	+03 30 46.618	11 19 17.523	11 19 17.809
1998 DU20	11 19 18.7762	+03 30 39.749	11 19 18.656	11 19 18.955
1998 DU20	11 20 10.3193	+03 25 31.635	11 20 10.802	11 20 10.480
1998 DU20	11 20 10.6784	+03 25 29.459	11 20 11.153	11 20 10.813
1997 NS7	11 19 44.2943	+03 27 09.199	11 19 44.725	11 19 44.314
1997 NS7	11 19 45.2125	+03 27 03.728	11 19 45.626	11 19 45.229
1997 NS7	11 20 27.2745	+03 22 56.676	11 20 27.741	11 20 27.000
1997 NS7	11 20 26.9833	+03 22 58.408	11 20 27.451	11 20 27.292

Asteroid Name	Measured DEC Before correction	Measured DEC After correction	ΔRA (arcsec) Before correction	ΔRA (arcsec) After correction
1998 DU20	+03 31 01.13	+03 30 45.88	1.81	2.47
1998 DU20	+03 30 52.85	+03 30 39.14	1.80	2.68
1998 DU20	+03 25 36.24	+03 25 30.68	7.23	2.41
1998 DU20	+03 25 34.32	+03 25 28.03	7.11	2.02
1997 NS7	+03 27 16.96	+03 27 09.22	6.45	0.29
1997 NS7	+03 27 10.34	+03 27 03.62	6.19	0.25
1997 NS7	+03 23 01.31	+03 22 58.47	6.99	0.25
1997 NS7	+03 23 02.72	+03 22 56.63	7.00	0.26

Asteroid Name	ΔDec (arcsec) Before correction	ΔDec (arcsec) After correction	Avg Off. (arcsec) Before correction	Avg Off. (arcsec) After correction
1998 DU20	14.51	0.74	**13.92**	**2.66**
1998 DU20	13.10	0.61	**13.92**	**2.66**
1998 DU20	4.60	0.96	**8.59**	**2.53**
1998 DU20	4.86	1.43	**8.59**	**2.53**
1997 NS7	7.76	0.02	**9.57**	**0.28**
1997 NS7	6.61	0.11	**9.57**	**0.28**
1997 NS7	4.63	0.06	**8.30**	**0.26**
1997 NS7	4.31	0.05	**8.30**	**0.26**

P1 (Pic ID): Frame unique identifier ID
PIC_FRAME_ID: Subaru frame ID
Pic Date: Date picture was taken
PIC_UT_STR : UT time picture was taken
Dist (arcsec): Dist. squared ΔRA & ΔDEC
Avg Offset : Average dist. for both frames
Horizons MAG: Absolute Magnitude returned by Horizons
Measured RA: Measured RA on frame using DS9
Measured DEC: Measured DEC on frame using DS9
ΔRA: Absolute Difference Horizons RA -Measured RA
Horizons DEC: Horizons DEC, center-site name: Mauna Kea
Horizons RA: Horizons RA, center-site name: Mauna Kea

For asteroid 1998 DU20, average offsets are 2.66 arc seconds on December 1 and 2.53 arc seconds on December 2 after frame corrections. For Asteroid 1997 NS7 average offsets are 0.28 arc seconds on December 1 and 0.26 arc seconds on December 2 after frame corrections. The differences between the offsets at the two dates are now well within the estimated measurement errors.

3.6.3- Calibration of remaining frames

We used the astrometry website [11] to process the remaining 692 frames. Some of these frames could not be calibrated or were rejected for the following reasons:

1- Some images were calibrated with a wrong pre-index hash. [34].
2- Some images could not to be corrected by the website for unknown reasons.
3- Pre-indexed hashes were not available in the catalog for some images.
4- Some images were of poor quality.

For any given asteroid, the offsets between measured and calculated positions on each frame should be about the same (See 3.6.1), therefore if a frame cannot be calibrated, we record the offset of the other frame without averaging both offsets.

If for any given asteroid, both frames cannot be processed then the offset of this asteroid is rejected.

The number of asteroids available on a pair of frames with at least one calibrated frame is: 419 asteroids + 42 asteroids = 461 distinct asteroids (See table 9). After this calibration process, we obtained the result summarized below:

Table 9 Identified ast. vs. calibrated/non-calibrated frames

Description	# of frames	# of distinct asteroids
Frames of poor quality and rejected	47	31
Both frames could not be calibrated and were rejected	24	16
One frame only could be calibrated	26	42
Both frames were successfully calibrated	595	419
Total:	**692**	**508**

3.6.4- Measured offsets and asteroid diameter results

We measured and calculated the average offsets of all the asteroids that have at least one calibrated frame (See table 9):

One frame only could be calibrated = 26 frames or 42 asteroids

Both frames were successfully calibrated = 595 frames or 419 asteroids

Thus, we measured and calculated the offsets of 419 + 42 = 461 asteroids. We then estimated the diameter of each asteroid.

3.6.4.1- Determination of asteroid diameters

Observations made by NASA's Wide-field Infrared Survey Explorer, or WISE, are available at [12]. The site provides diameter and albedo data of some 139,481 asteroids. 49 out of 461 asteroids were found in this survey and the corresponding diameters were used in our data. For the remaining asteroids, we estimated the diameters, from the absolute magnitude H of the asteroid (the intrinsic brightness of the asteroid at 1 AU from Earth and 1 AU from the Sun)

and its albedo p according to the equation 1 below:

$$D_{(km)} = \frac{1329}{\sqrt{p}} 10^{-0.2H} \qquad \text{(Equation 1)}$$

The absolute magnitude H is taken from the Lowell database (Fields f4, see 2.5: "The asteroid orbital elements table"). We use the average albedo p = 0.15.
H is typically uncertain to about 0.1 mag., this gives an uncertainty on the calculated diameter of about 5%. Sometime, the uncertainty on the albedo could be as much as 50% giving an uncertainty on diameter of 25%.
From this result, the smallest identified asteroid is 2009 WD106 with a diameter of 0.64 km. The largest one is 2003 YW179 with a diameter of 148 km. The average size of the 461 asteroids is 3.4 km. A sample of 6 asteroids with calculated diameters or diameters taken from WISE when available is shown on table 10. The column "Best diameter" is the one we used in the result and discussion that follow. The detailed result is given is in appendix E: List of identified asteroids.

Table 10 Identified ast. with their defined or calculated diameters

Ast ID	Asteroid Name	Abs. Mag H	Avg offset (arcsec)	WISE Diameter (km)	Calc. Diameter (km)
33320	1998 OP12	14.7	1.36		3.9
11068	1992 EA	14.7	1.23		3.9
55382	2001 SS265	14.7	0.43	3.518	
86474	2000 CT79	14.9	0.44	2.88	
95551	2002 EB93	14.9	0.25		3.6
87107	2000 LD20	15	0.17		3.4

3.6.4.2- Correlation between offsets and diameters

When we plot the average measured offset with the diameter, we find that the offsets of asteroids smaller than 3 km are significant. (See figure 10). The average measured offsets are between 0.03 arc seconds and 86.8 arc sec.

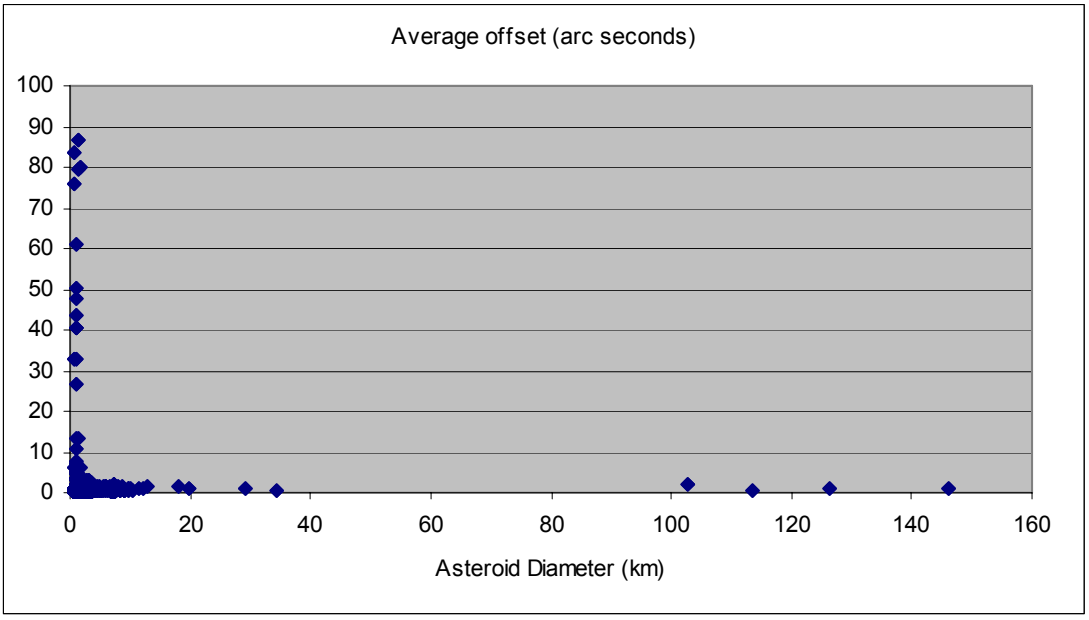


Figure 10: Sample of asteroids; measured offsets plot against diam.

3.6.4.3- Correlation between measured RA / Dec and diam.

When we plot the relative average measured ΔRA and ΔDec offsets (± 0.65 arcsec, see section 4.1: Error analysis), with the diameter. We find that there is no preferential offset direction. (See figures 11, 12, 13 and 14).

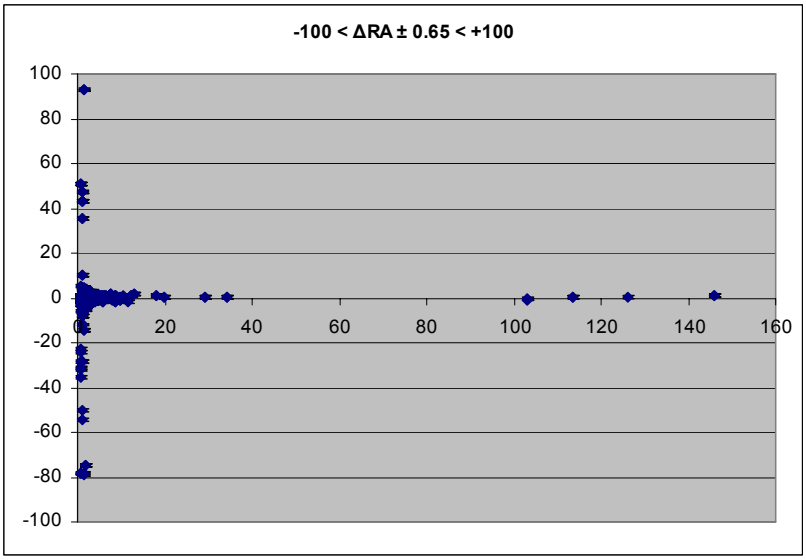


Figure 11: Measured |ΔRA| ≤ 100 arcseconds vs. asteroid's diam.

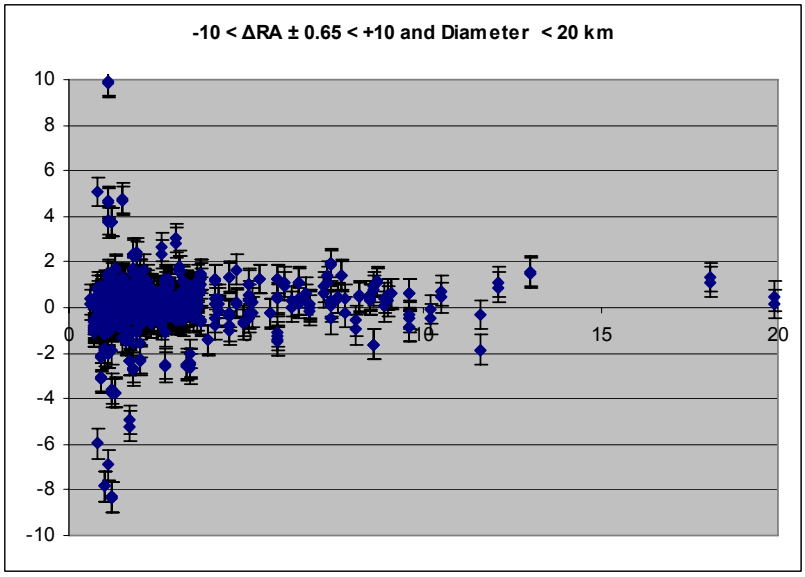


Figure 12: Measured |ΔRA| ≤ 10 arc seconds vs. asteroid's diam.

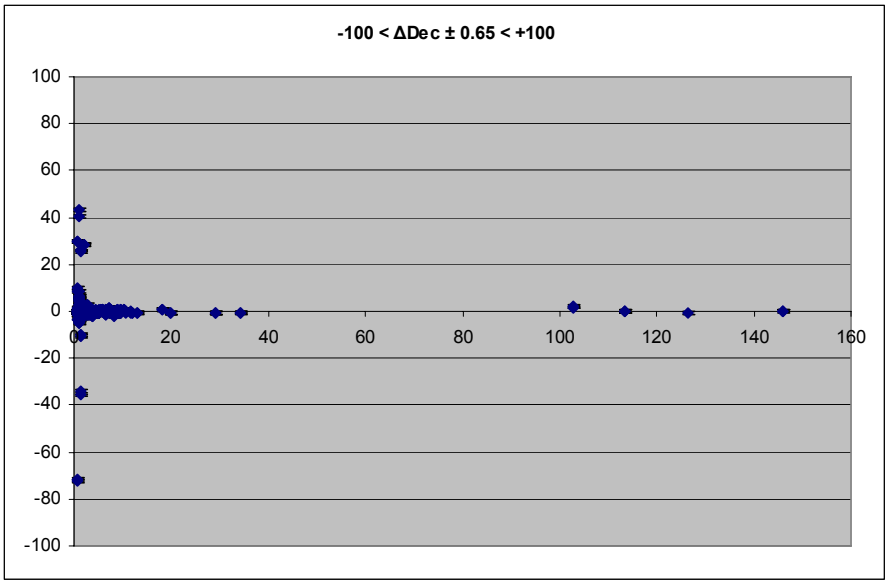


Figure 13: Measured |ΔDec| ≤ 100 arc seconds vs. asteroid's diam.

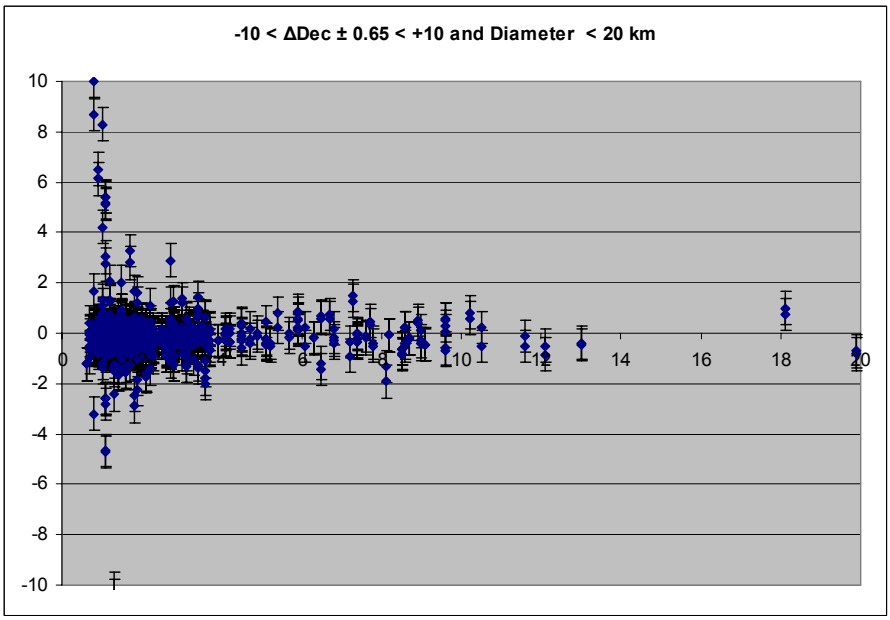


Figure 14: Measured |ΔDec| ≤ 10 arc seconds vs. asteroid's diam.

The fact that the measured RA and DEC offsets are larger for smaller asteroids would be consistent with the expected trend from the Yakovsky effect, but as we explain in section 4.2.5: "Comparison with measured offsets", the offsets are too large.

CHAPTER 4- DISCUSSIONS

4.1- Error analysis

The resolution of the Subaru prime-focus camera is 0.202 arc seconds per pixel [33]. The measurement uncertainty is about ± 2 pixels, giving a measured offset uncertainty:

$$\Delta P = \sqrt{(0.404)^2 + (0.404)^2} = \pm 0.571 \text{ arc sec.} \quad \text{(Equation 2)}$$

After each frame was calibrated, the frames within a pair of frames should show the same measure offsets. However, there is an average offset difference of ± 0.11 arc seconds between each frame within a pair. Additionally, we evaluate a frame calibrations or DS9 error of ± 0.5 arc seconds.

$$\Delta RA = \Delta Dec = \sqrt{(0.404)^2 + (0.11)^2 + (0.5)^2} = \pm 0.652 \text{ arc sec.} \quad \text{(Equation 3)}$$

$$\Delta M = \sqrt{(0.571)^2 + (0.11)^2 + (0.5)^2} = \pm 0.767 \text{ arc sec.} \quad \text{(Equation 4)}$$

Thus, we have an average error bar of ~ ± 0.77 arc seconds on our offset measurements and ~ ± 0.65 arc seconds for our RA and Dec measurements. We apply this offset uncertainty to the plot given in figures 15 and 16.

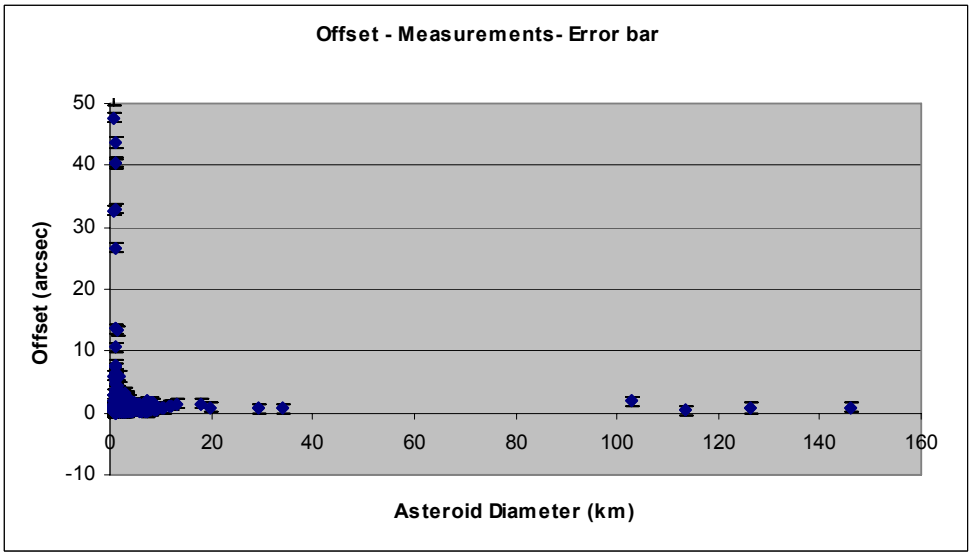


Figure 15: Offset measurement error-bars plot (offset ≤ 50 arc sec.)

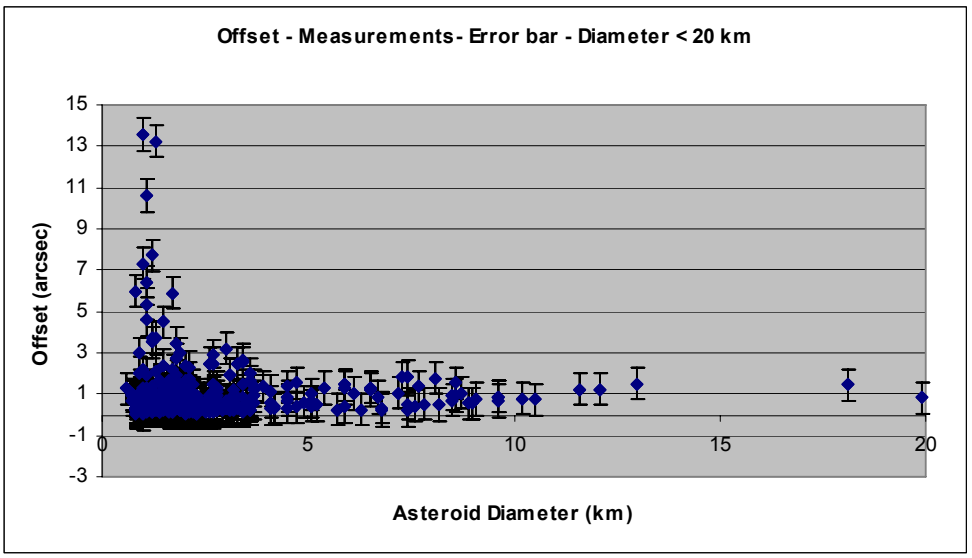


Figure 16: Offset measurement error-bars plot (offset ≤ 15 arc sec.)

The measured offsets seem to increase as the asteroid get smaller which is also consistent with the expected trend of the Yakovsky effect. However, the magnitude of these offsets is much greater than the Yarkovsky drift calculation as we are going to see in the next section.

4.2- Yarkovsky drift calculations

The Yarkovsky acceleration equations are given in the Bottke et al paper [23]. (We also give the detail theory of the Yarkovsky Effect in appendix A). The two diurnal and seasonal Yarkovsky drift equations are:

$$\left[\frac{da}{dt}\right]_{Seasonal} = \frac{4\alpha}{9}\cdot\frac{\Phi}{n}\cdot F_{\nu,p}(R',\Theta)\cdot\sin^2\gamma \qquad \text{(Equation 5)}$$

$$\left[\frac{da}{dt}\right]_{Diurnal} = -\frac{8\alpha}{9}\cdot\frac{\Phi}{n}\cdot F_{\nu,p}(R',\Theta)\cdot\cos\gamma \qquad \text{(Equation 6)}$$

The total calculated drift is simply the linear superposition or the sum of both effects. We did this calculation for 10 known asteroids whose semi-major axis drifts have been either measured by radar or previously calculated by others. We then compared our results with these other results in order to validate the method used. The comparison between both results is given in sections 4.2.3 and 4.2.4.

$\Phi = \frac{\pi R^2 \xi_o}{mc}$ with Φ the radiation pressure coefficient and with: (Equation 7)

ξ_o the solar radiation flux

R the radius of the asteroid

m the mass of the asteroid

c the light velocity

n the mean motion

ω the angular frequency

γ the obliquity of the spin axis

Θ the thermal parameter

$R' = \frac{R}{l_\upsilon}$ Is the scaled radius

l_ν the penetration depth (thermal wave)

ν = n (mean motion for seasonal effect), ν =ω (angular freq. for diurnal effect).

$$F_{\nu,p} = \left[\frac{\kappa_1(R')\Theta_p}{1 + 2\kappa_2(R')\Theta_p + \kappa_3(R')\Theta_p{}^2} \right] \qquad \text{(Equation 8)}$$

is the thermal function that depends on R', Θ_p.

p represents the order of a Fourier expansion used in the theory, in this expansion and for simplification, we only had considered the first term (p = 1).

Using this Fourier expansion, we found an expression for Θ after development to the first order: $\Theta_1 = \frac{K}{4\sqrt{2}\varepsilon\sigma \cdot l_\nu \cdot T_o^3}$ (Equation 9)

We also found an expression of the surface temperature and the penetration depth of the thermal wave:

$$T_o = \left[\left[\frac{a_o}{a} \right]^2 \cdot \left[\frac{(1-A)\xi_o}{4\varepsilon\sigma\sqrt{1-e^2}} \right] \right]^{1/4} \qquad \text{(Equation 10)}$$

$$l_\nu = \sqrt{\frac{K}{\rho C_p n}} \quad . \qquad \text{(Equation 11)}$$

With:

A the albedo of the asteroid

e the eccentricity of the asteroid

a the semi-major axis of the asteroid

a_o the distance Sun-Earth (AU)

ε the emissivity of the asteroid

σ the Stephan Boltzman constant

C_p the specific heat of the asteroid

ρ the density of the asteroid

K the thermal conductivity

l_ν the penetration depth (themal wave)

The eccentricity and the semi-major axis are part of the orbital element and always available. The mass m is often available. When the radius is not given we deduce it from the absolute magnitude (See determination of asteroid diameters in 3.6.4.1).

The angular frequency ω is obtained from the rotation period P given often in hours. The relationship between the rotation period of the asteroid and the angular frequency is defined as:

$$\omega = \sqrt{2\pi \frac{P}{3600}} \qquad \text{(Equation 12)}$$

The mean motion n is calculated from: $n = \sqrt{G \frac{(M+m)}{a^3}}$ (Equation 13)

or from the orbital period when it is available. G is the gravitational constant, M the mass of the Sun, m the mass of the asteroid (negligible!) and a is the semi-major axis of the asteroid.

When parameters are not available, we use the following values:

Albedo A: We take the average albedo value A = 0.15.

Emissivity ε: We use 0.9.

Rotation period: We use the average rotation period of 4 hours.

When the thermal conductivity K is not available, we use the thermal inertia method published by [13] and showing an inverse correlation between the size of an asteroid and its thermal inertia following the empirical formula:

$\Gamma = d_0 D^{-\xi}$ (Equation 14)

With: $\xi \sim 0.4$ and $d_0 \sim 300$ (km).

The Thermal inertia is also defined as: $\Gamma = \sqrt{K\rho C_p}$ (Equation 15)

Thus, we can make reasonable assumptions for K and C_p.

When the density ρ is not available we take the average value ρ = 2500 kg·m^3.

The predominance of negative drift rates in this sample of asteroids is consistent with the presumed delivery method of near-Earth asteroids from the main belt.

To calculate the drift acceleration in km from the one given in AU/My, we used the method published by [14]. It gives the following orbital shift:

$$\Delta\rho \approx 7\dot{a}_4(\Delta_{10}t)^2 a_{au}^{-3/2} \text{ km}$$ (Equation 16)

$\dot{a}_4$: The Yarkovsky drift in 10^{-4} AU/Myr

$\Delta_{10}t$: The time difference in tens of years

a_{au} : The semi-major axis of the asteroid in AU

The calculated drift rates da/dt and Δρ for 5 NEO's are given below in table 11.

(See also for more details appendix E: List of identified asteroids).

Table 11 Sample of 5 NEO's with their calculated Yarkovsky drift

#	Asteroid Name	Dia. (m)	Dens. ($m.kg^{-1}$)	Da/dt ($x10^{-4}$ AU/My) (calculated)	# yrs Obs.	S.-major Axis (AU)	Sea. Δρ (km)	Diurnal Δρ (km)	Σ Drift Δρ (km)
1	1992 BF	401	2500	-8.55	52	0.9079552	-17.6	-1851.9	-1869.5
2	Golevka	530	2700	-12.04	16	2.4983420	-8.3	-46.3	-54.6
3	Apollo	1500	2700	-2.87	78	1.4700910	-7.6	-677.2	-684.9
4	Bacchus	630	500	-10.04	30	1.0782008	-67.2	-497.9	-565.1
5	Yorp	105	2500	-33.41	5	1.0058757	-0.02	-57.9	-58.0

4.2.1- The thermal inertia dependence

The thermal inertia is the resistance of a material to temperature change. It is defined by: $\Gamma = \sqrt{K\rho C_p}$ (See equation 15) where K is the thermal conductivity, ρ the density and C_p the specific heat. When the thermal inertia is smaller than 10 or very high (larger than 2000), the strength of the Yarkovsky effect approaches zero. Typical values of thermal inertia are 50 for the Moon and between 20 and 400 for main belt asteroids [35].

The thermal parameter: $$\Theta_1 = \frac{K}{4\sqrt{2}\varepsilon\sigma \cdot l_\nu \cdot T_o^3}$$ (Equation 17)

and the penetration depth of the thermal wave $l_\nu = \sqrt{\frac{K}{\rho C_p \nu}}$ can be rewritten as:

$$\Theta_1 = \frac{K}{4\sqrt{2}\varepsilon\sigma \cdot T_o^3} \cdot \sqrt{\frac{\rho C_p \nu}{K}} = \frac{1}{4\sqrt{2}\varepsilon\sigma \cdot T_o^3} \cdot \sqrt{K\rho C_p \nu} = \frac{\Gamma\sqrt{\nu}}{4\sqrt{2}\varepsilon\sigma \cdot T_o^3}$$ (Equation 18)

Thus, we can replace the dependence of the Yarkovsky effect on thermal conductivity K and specific heat C_p by a dependence on the thermal inertia only. This is an important dependence when we use the theory to calculate the orbital drift of an asteroid. Often, K and C_p are not available. The thermal inertia Γ can be, in this case, deducted from the the empirical formula: $\Gamma = d_0 D^{-\xi}$. The expression of the thermal parameter would be in this case:

$$\Theta_1 = \frac{\Gamma\sqrt{\nu}}{4\sqrt{2}\varepsilon\sigma \cdot T_o^3}$$ (With: ν = n for diurnal and ν = ω for seasonal) (Equation 19)

Figure 17 below shows the calculated strength of the Yarkovsky drift for asteroid Yorp (54509) versus the thermal inertia Γ. ($0 \le \Gamma \le 2000$).

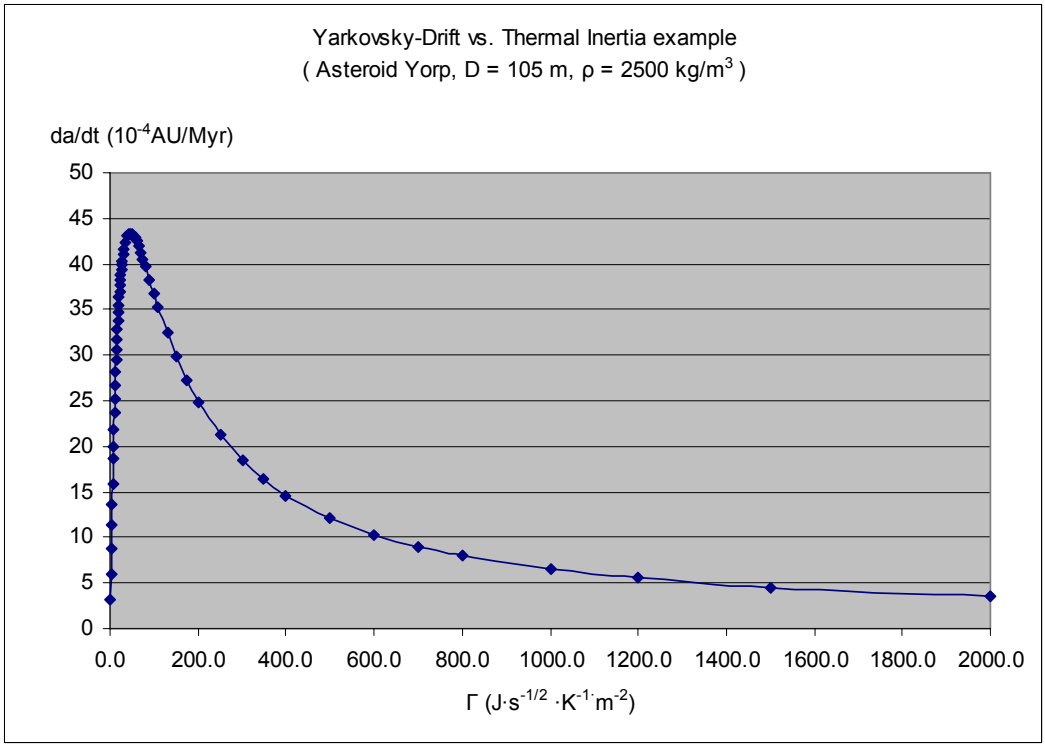


Figure 17: Yorkovsky drift vs. thermal inertia

4.2.2- Comparison between diurnal and seasonal effects

It is also interesting to compare the strength of both diurnal and seasonal Yarkovsky effects with the obliquity γ. The strength of the diurnal effect is always more important except for an obliquity in the 85^{o} -100^{o} range. Furthermore, the seasonal effect is always negative so even if its effect is weaker, the average effect on large sample of asteroids causes this component to be significant. On the other hand, the average effect of the diurnal component of the Yarkovsky effect is equal to 0. So for statistical analysis, only the seasonal component needs to be considered. However, for an individual calculation, the diurnal effect has a better chance to be detected. (See below figure 18).

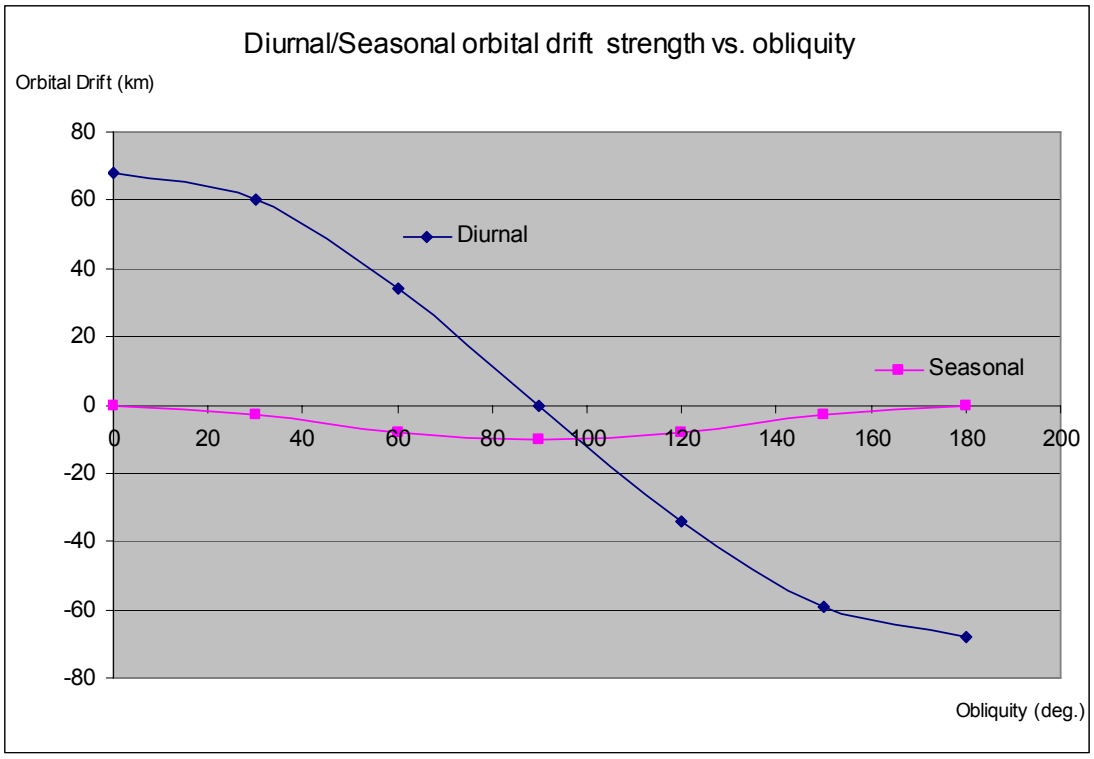


Figure 18: Estimated Orbital drifts for an asteroid of diameter 100 m

4.2.3- Comparison with previous calculations

A direct calculation of the Yarkovsky effect was published by [15]. The extract from page 1 is given below in table 12.

Table 12 Some published calcul. asteroid's Yarkovsky orbital drift

Object	Da/dt ($x10^{-4}$ AU/My)	SNR	Observed Arc	Radar Apparition	Absolute Magnitude
152563 (1992 BF)	-10.78 ± 0.73	14.8	1953-2005	N/A	19.7
6489 Golevka	-6.39 ± 0.44	14.4	1991-2007	1991, 1995, 2003	19.2
1862 Apollo	-2.44 ± 0.26	9.3	1930-2008	1980, 2005	16.3
2063 Bacchus	-10.59 ± 2.21	4.8	1977-2007	1998	17.1
54509 Yorp	-25.12 ± 6.18	4.2	2000-2005	2001, 2004	22.6
2340 Hathor	-13.94 ± 3.84	3.6	1976-2007	N/A	19.2
101955 (1999 RQ36)	-15.69 ± 4.99	3.1	1999-2006	1999, 2005	20.8
1620 Geographos	-1.18 ± 0.39	3.0	1951-2008	1983, 1994	15.6

(http://astro.mff.cuni.cz/davok/papers/Chesley_etal_ACM08.pdf)

The published results are then compared with the Yarkovky calculation of the orbital drifts of these 8 NEO's (diurnal + seasonal) to validate our method. (See table 13).

Table 13 Calcul. Yarkovsky drifts, difference with published values

#	Asteroid Name	Dia. (m)	Dens. $kg \cdot m^{-3}$	Rotation Period (hr)	Da/dt ($x10^{-4}$ AU/My) (published)	Da/dt ($x10^{-4}$ AU/My) (calculated)	Diff. with Publ. val.
1	1992 BF	401	2500	4.00	-10.78	-8.55	20.7%
2	Golevka	530*	2700	6.03	-6.39	-12.04	88.4%
3	Apollo	1500*	2700	3.07	-2.44	-2.87	17.6%
4	Bacchus	315*	500	14.90	-10.59	-10.04	5.2%
5	Yorp	105	2500	0.20	-25.12	-33.41	33.0%
6	Hathor	380*	2500	4.00	-13.94	-9.76	30.0%
7	1999 RQ36	408	1400	4.29	-15.69	-18.66	19.9%
8	Geographos	2560*	2500	5.22	-1.18	-1.28	8.5%
						Average:	27.8%

* Diameter given. (not calculated from absolute magnitude).

The detail of these calculations is given in Annexe F: Yarkovsky calculations, examples (See example 1). The average difference between calculations and published values is ~28%. This difference is possibly caused by the limit in the order 1 used in the theory. See Annexe A: Detail theory of Yarkovsky effect, 1.12 evaluation of (E, ΔT), equations 37 and 38.

4.2.4- Comparison with radar orbital shift measurements

Orbital shifts have been so far directly observed and measured with radar observations for only two asteroids: Golevka (6489) and 1992 BF (152563). Paper published by [16].

Asteroid Golevka has been observed via radar in 1991, 1995, 1999 and 2003. Over these 12 years, a shift of 15 km in the range was detected [14, 17].

Asteroid 1992 BF has been observed with astrometry from 1953 to 2005. Over these 52 years, a $\frac{da}{dt}$ of 10.7×10^{-4} AU/Myr was detected (2,341km) [18].

This shows that the Yarkovsky effect has to be evaluated when the dynamical behavior of an asteroid needs to be studied. For example, if we want to calculate

the probability of an asteroid colliding with the Earth, a correction of its drift by the Yarkovsky effect should be carefully considered.

4.2.5- Comparison with measured offsets

We adopt the calculation method described in section 4.2: "Yarkovsky drift calculations" to evaluate the drift of a generic asteroid whose orbital elements, semi-major axis and eccentricity are the average of the 461 asteroids we identified in the Subaru pictures. We chose an average thermal inertia of $300\ J \cdot m^{-2} \cdot J \cdot s^{-0.5} \cdot K^{-1}$ (from: $\Gamma = d_0 D^{-\xi}$), an average density of 2,500 $kg \cdot m^{-3}$, an average albedo of 0.15, an average rotation period of 4 hours and an average emissivity of 0.9. We also chose an obliquity $\gamma = 180^o$ because this gives an optimal diurnal only drift. For $\gamma = 180^o$, the seasonal effect is zero and the diurnal effect is maximum.

The diameters vary like our sample from 0.6 km to 140 km. The corresponding calculated drifts, over a period of 6 years, for this generic asteroid are shown in table 14 and the corresponding plot is shown in figure 19. The calculated drift plot is similar in shape and trend to the one given in figure 10 and seems to indicate that a Yarkovsky effect is the likely source of these measured drifts because both plots have similar shapes. However, the scales of the offsets in each plot are very different. We also notice on the logarithmic plot (figure 19), that the drift is proportional to the inverse diameter.

Table 14 Calculated Yarkovsky for a generic ast.: drifts vs. diameters

Asteroid Diameter (km)	Drift (theory) (km)	Asteroid Diameter (km)	Drift (theory) (km)
0.6	7.39	2.6	1.7
0.8	5.54	4	1.11
1	4.43	6	0.74
1.2	3.69	10	0.44
1.4	3.17	20	0.22
1.6	2.77	40	0.04
1.8	2.46	60	0.03
2.2	2.01	100	0.02
2.4	1.85	140	0.01

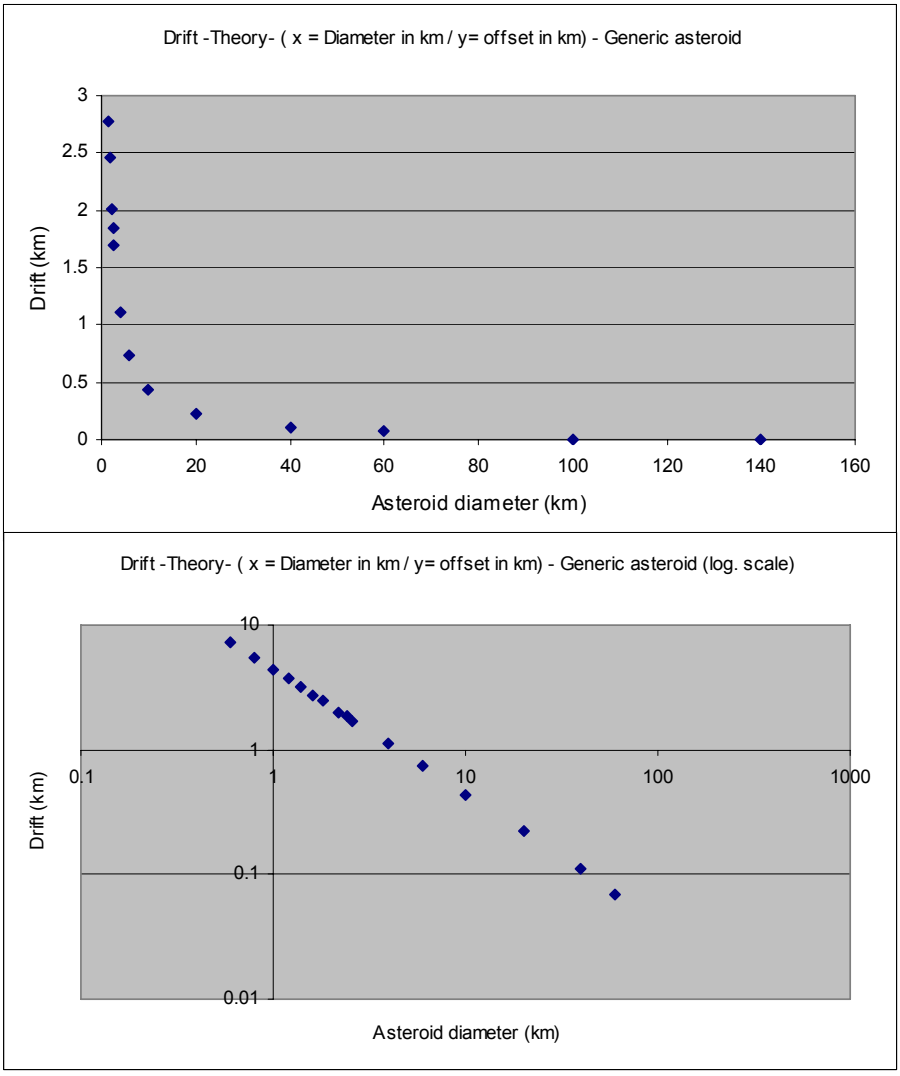


Figure 19: Plot of a generic asteroid; calcul. orbital drifts vs. diam.

To confirm this, and using again the same calculation method described in section 4.3, we calculated the theoretical Yarkovsky orbital shifts of 6 of the 461 identified asteroids from our Subaru sample. We picked the following identified asteroids:

Hypnos (14827)	2009 RG35 (467065)
2000 TV35 (242212)	2005 YU17 (345659)
2009 WD106 (476503)	2005 WV48 (342421)

We chose these six asteroids because their measured offsets are small, because 2009 WD106 and 2005 WV48 are the smallest asteroids in the sample, and also because the orbital and physical data of these asteroids are confirmed and more certain:

Hypnos (f11 = 154), 2009 RG35 (f11 = 26), 2000 TV35 (f11 = 48)

2005 WV48 (f11 = 54), 2009 WD106 (f11 = 77), 2005 YU17 (f11 = 22)

(See section 2.5: The asteroid orbital elements table, f11= Number of observations used in orbit computation).

Finally, asteroids 2009 RG35, 2009 WD106 and 2000 TV35 are on the list of asteroids which have been identified on pictures that pre-date the year of their confirmed discovery. (See section 3.5: identified asteroid results).

We defined the number of years observed as the interval of time between the date of the picture and the date of submission to Horizons.

We chose an obliquity of $\gamma = 180^{o}$ to maximize the offset.

The calculated offset for these 6 asteroids is given below in table 15.

Table 15 Calculated Yarkovsky orbital drifts of 6 identified asteroids

#	Asteroid	Dia. (m)	Dens. (kg/m^3)	Da/dt ($x10^{-4}$ AU/My) Calculated - Max.	# yrs Obs.	S.-major Axis	Diurnal $\|\Delta\rho\|$ (km)	Σ Drift $\|\Delta\rho\|$ (km)
1	Hypnos	751	2500	-14.89	6	2.8428697	-7.83	7.83
2	2009 RG35	1046	2500	-8.03	6	2.6627106	-4.35	4.35
3	2000 TV35	2030	2500	-3.42	6	2.7990602	-1.84	1.84
4	2005 YU17	801	2500	-10.81	6	2.3451229	-7.59	7.59
5	2009 WD106	642	2500	-15.59	6	1.9343860	-14.61	14.61
6	2005 WV48	660	2500	-13.59	6	2.2206098	-10.35	10.35
							Average Δρ ~	7.76 km

We used the average rotation period of 4 hrs.

To evaluate the measured drift of these same asteroids, we convert the angular offset ΔΘ (in arc seconds) in distance Δr (in km) using the conversion formula:

$$\Delta r = D \cdot \tan(\Delta\Theta) \quad .. \quad \text{(Equation 20)}$$

D is the distance Earth - asteroid (in km).

This distance D (AU) is obtained from the Horizons system [5]. (Parameter #20 or parameter delta).

The measured offsets obtained for the same 6 asteroids are shown in table 16.

Table 16 Measured Yarkovsky offsets of a sample of 6 identified ast.

Asteroid	Picture Date	Picture Time (UT)	Offset ΔΘ (arcsec)	D (AU)	D (km)	Offset (AU)	Offset Δr (km)
Hypnos	08/30/05	13:48:53.117	0.177658	1.5553	2.33E+08	1.34E-06	200
2009 RG35	08/30/05	14:17:03.409	0.032176	2.4936	3.73E+08	3.89E-07	58
2000 TV35	08/30/05	13:42:52.331	0.059023	2.1132	3.16E+08	6.05E-07	90
2005 YU17	08/30/05	12:36:25.780	0.083203	2.2224	3.32E+08	8.96E-07	134
2009 WD106	01/05/05	10:00:24.362	1.257979	1.8374	2.75E+08	1.12E-05	1,676
2005 WV48	08/30/05	14:11:02.585	1.008348	1.8197	2.72E+08	8.90E-06	1,331
						Avg Δr ~	582 km

The orbital drift Δρ or semi-major axis drift of a given asteroid manifests itself with a measured offset Δr. If the orbital drift of this asteroid is all radial, then the measured offset Δr will be measured as zero. In contrast the maximum measured offset will be equal to the orbital drift. We can modelize this with a spherical coordinate's reference where we consider the projection of the orbital drift Δρ on the offset Δr such as $\Delta r = \Delta\rho \cdot \sin\Theta$.

On average, the orbital drift Δρ in semimajor axis will yield an offset:

$\Delta r = 0.64 \cdot \Delta\rho$ (Equation 21)

We can use this Δr/Δρ ratio to estimate the average calculated offset of these 6 asteroids. The average calculated orbital drift for these 6 asteroids is: Δρ = 7.76 km or Δρ ~ 8 km (see table 15) which gives an average offset of:
Δr = Δρ x 0.64 = 4.96 km or ~ 5 km. The average measured offset is 582 km (see table 16) which gives a ratio between calculated and measured offsets of ~ 116.

Such a ratio rules out a possible Yarkovsky effect as the cause of the measured offsets in our sample of 461 asteroids. (Shown in figure 10).

To explain this greater measured offset for smaller asteroids, a possible other reason (besides the Yarkovsky one that has been ruled out) could be a greater imprecision in their orbital elements: The smaller an asteroid, the harder it is to make sufficient observations in order to obtain precise enough orbital elements.

One way to verify this hypothesis is to plot the measured offset against the parameter f11 or "the number of times an asteroid has been observed". (See in 2.5: "The asteroid orbital elements table"). The result is shown in figure 20. The plot shows that the measure offsets are larger for smaller values of f11.

We also plotted the measured offset against the parameter f10 or "the orbital arc, days, spanned by observations". (See also 2.5: "The asteroid orbital elements table"). The result is shown in in figure 21. The plot shows that the measure offsets are also larger for smaller values of f10.

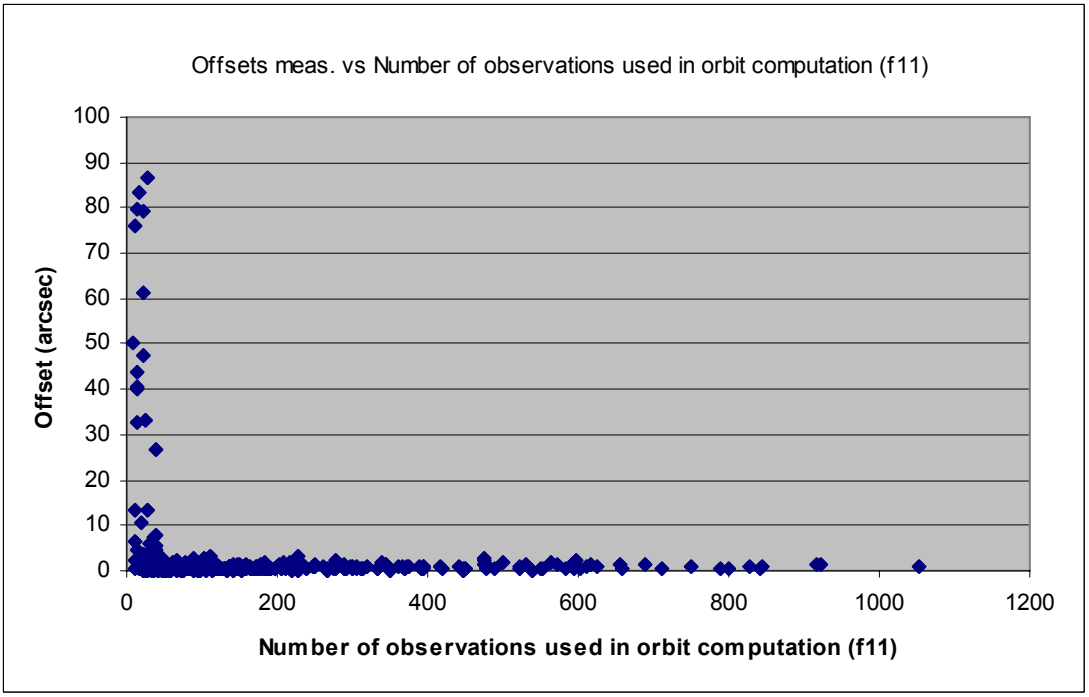


Figure 20: Plot of the 461 ast.: Measured offsets vs. # of observ.

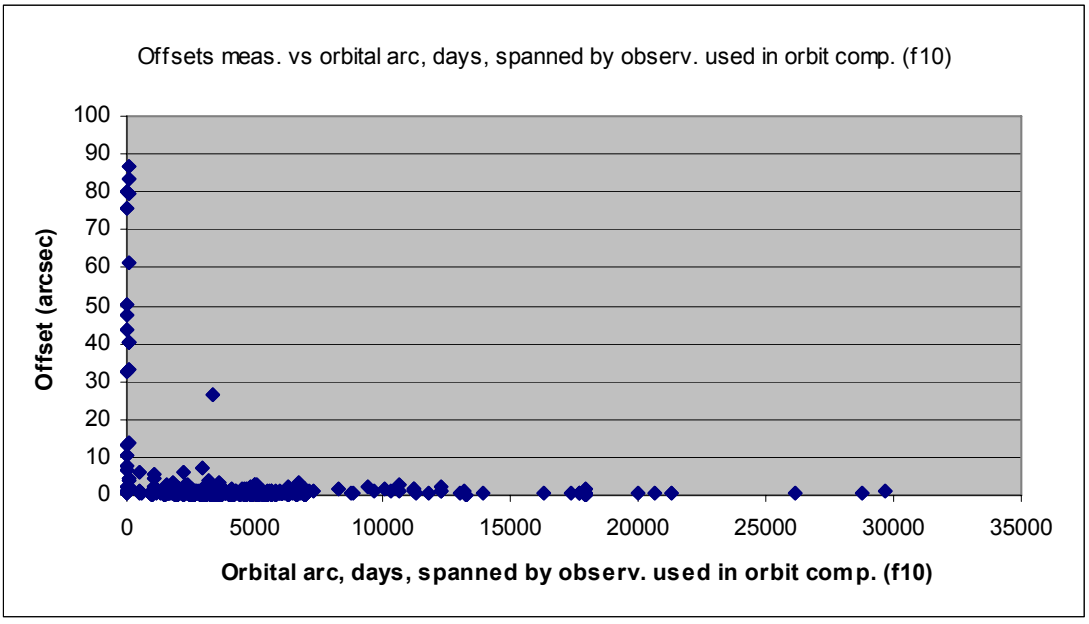


Figure 21: Plot of the 461 ast.: Measured offsets vs. orb. arc, days

CHAPTER 5- CONCLUSION

One of the best innovations during this research was the ability to design a database system that successfully predicts and identifies 508 serendipitous asteroids on archived astronomical imaging. In addition, we found that there is a strong correlation between their measured offsets and their sizes. However, we could not detect any Yarkovsky effect during our statistical analysis or by comparing the theory with the measurements for the three following reasons:

First, when we plot the relative average measured RA offsets of the entire sample of asteroids against their diameter, we found that there is no preferential offset direction ruling out the possibility of detecting a Yarkovsky effect using this statistical analysis method. Second, the average ratio between the measured offsets of 6 of some of the smallest asteroids in our sample and their calculated offsets gives a ratio of 116. Third, the measured error we identified (± 0.77 arc sec) gives an average measured offset error of ± 1600 km and an average orbital drift error of ± 2500 km (using the $\Delta\rho/\Delta r$ ratio). By comparison, the average diameter of the asteroids in our sample is 3.6 km and the smallest one has a diameter of 600 m. This smallest asteroid would cause a theoretical orbital drift of less than 15 km over 6 years. This drift is too small to be detected by a measurement method that has an average orbital drift error of ± 2500 km. These last two points rule out completely the possibility of detecting a Yarkovsky effect in our sample by comparing theory and measurements on astronomical imaging.

We also found that the plot showing the measured offsets against the number of observations indicates a strong correlation between larger offsets and fewer numbers of observations similar to the correlation we found between measured offsets and sizes. This strong correlation seems to indicate that the ephemeris becomes more inaccurate as the size of the asteroids gets smaller. The fainter an asteroid, the more difficult it is to obtain accurate ephemeris and this could be the same reason that would explain the correlation between measured offsets and sizes that we found in our sample.

For future work, the design of this database system and its method could be used to identify fainter asteroids whose ephemeris needs to be corrected.

In addition, since we could identify 96 asteroids on pictures that pre-date the year of their confirmed discovery, this same method could also be used to identify new asteroids on existing astronomical imaging.

Finally, extending this method to older astronomical imaging, identifying smaller asteroids with very well defined orbital elements, and increasing the sample of asteroids might also allow us to establish a correlation between measured offsets and Yarkovsky effect as long as statistical analysis is done.

CHAPTER 6- SUMMARY

The major results of this study can be summarized as follows:

1. We successfully wrote a database system and a program that predicted and identified 508 serendipitous asteroids in 692 astronomical frames taken from 2004 to 2009, including one NEO and 96 asteroids identified on pictures that pre-date the year of their confirmed discovery.

2. The finding of 26 asteroids identified in multiple pictures and the comparison of their offsets allowed us to realize that these 692 frames needed to be calibrated in order to make consistent offset measurements. This calibration process allowed us to measure the offsets of 461 distinct asteroids.

3- We reviewed the detailed mathematical formulation of the seasonal and diurnal Yarkovsky effect. We described a method to compute this effect and we applied it to some well known NEO's. We compared our results with other published ones and found a good agreement between the two sets.

4- We computed several statistics to try to identify a possible Yarkovsky effect on our sample of 461 distinct asteroids. The statistical results were not conclusive because they did not show any Yarkovsky effects.

5. We could successfully demonstrate that there is a strong correlation between the size of the asteroids studied in our sample and their offsets. When plotted, the overall distribution of the offsets of the asteroids relative to their size is very similar to the one we obtained when we use the theory.

6- We used the same mathematical method to evaluate what would be the Yarkovsky effect of some of the asteroids that belong to our studied sample but we could not correlate these theoretical results with their measured offsets. The measured offsets are much larger than the ones predicted by the theory (by a factor of 50). The error analysis confirms that the Yarkovsky effect is too small to be detected.

7- Finally, we plotted the measured offsets vs. the number of observations for our sample of asteroids. We also plotted the measured offsets vs. the orbital arc, days, spanned by observations used in orbit computation. These two plots seem to indicate that the lack of precise orbital information in the ephemeris is a reasonable explanation for the strong correlation found between the sizes of the asteroids studied in our sample and their measured offsets.

APPENDIX A: DETAIL THEORY OF YARKOVSKY EFFECT

The Yarkovsky acceleration equations are given in the Bottke et al paper [23]. This appendix gives step-by step calculations that lead to the formulation of the following two diurnal and seasonal Yarkovsky equations:

$$\left[\frac{da}{dt}\right]_{Seasonal} = \frac{4\alpha}{9}\cdot\frac{\Phi}{n}\cdot F_{n,p}(R',\Theta)\cdot\sin^2\gamma \qquad \text{(Equation 22)}$$

$$\left[\frac{da}{dt}\right]_{Diurnal} = \frac{8\alpha}{9}\cdot\frac{\Phi}{n}\cdot F_{\omega,p}(R',\Theta)\cdot\cos\gamma \qquad \text{(Equation 23)}$$

1.1 Energy dE of solar radiation hitting an area dA during a time dt can be written as:

$$\boxed{dE = \xi\cdot\hat{s}\cdot\hat{n}\cdot dA\cdot dt} \qquad \text{(Equation 24)}$$

$\hat{s}$: Unit vector toward the sun

$\hat{n}$: Unit vector normal to the area dA

ξ : Solar radiation flux

1.2 Force df produced by the solar radiation pressure acting on an area dA is:

$$\boxed{\vec{df} = -\frac{1}{c}\cdot\frac{dE}{dt}\cdot\hat{s} = \frac{\xi}{c}\cdot\hat{s}\cdot\hat{s}\cdot\hat{n}\cdot dA} \qquad \text{(Equation 25)}$$

c: Light velocity

1.3 The power released (or radiated) per unit area for a black body is:

$\boxed{P = \sigma\cdot T^4}$ (Stefan Boltzman law) (Equation 26)

T : Temperature of the surface body

σ : Stefan Boltzman constant = $5.67\cdot10^{-8}$ $W\cdot m^{-2}\cdot K^{-4}$

1.4 The power radiated by a real body (asteroid) is:

$$\boxed{P\alpha = \varepsilon\sigma \cdot T^4}$$ (Equation 27)

ε: Emissivity of the body $0 \leq \varepsilon \leq 1$

$\alpha = 1-A$ α is the albedo-factor $0 \leq \alpha \leq 1$ with:

A is called the "Bond Albedo" also the fraction of power in the radiation incidents on the asteroid scattered back out into space.

1.5 Energy of a particle E and momentum p:

$E = mc^2$ and $p = mc \Rightarrow E = pc$

Thus:

$$\boxed{E = pc}$$ (Equation 28)

1.6 Power radiated from the surface of a body and transferred away from this surface:

$$\boxed{\frac{P}{c} = \frac{\sigma\varepsilon T^4}{c} = \frac{dp}{dt}}$$ (Equation 29)

(conservation of momentum)

1.7 Amount of power P emitted perpendicularly to the surface of a body

The Conservation of heat flux is: Power emitted = Incoming sunlight Power - Power of sunlight reflected

Or:

$$P = \frac{\partial Q_{in}}{\partial t} - \frac{\partial Q_{out}}{\partial t} = \int_{\Omega} \vec{\xi}_e \cdot \overrightarrow{dA}$$

ξ_e : Heat flux emitted by the body (asteroid)

Q : Heat

P : Energy transferred per second or Power

Ω : Surface to integrate from

For Ω being a hemisphere and considering a ring dA from this hemisphere, we get:

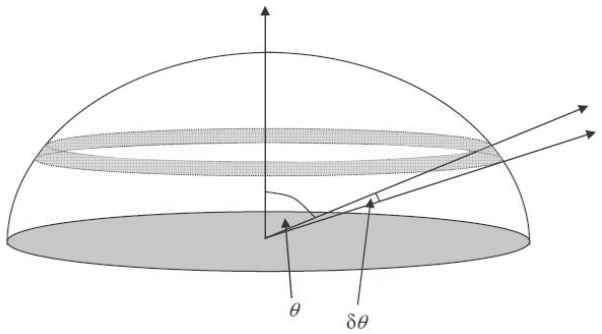
dA (ring) = $\sin\theta \cdot \sin\varphi \cdot d\varphi \cdot d\theta = 2\pi \cdot \sin\theta \cdot d\theta$

$\varphi : 0 \Rightarrow 2\pi$

$\theta : 0 \Rightarrow \frac{\pi}{2}$

Incoming sunlight

↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓



The heat flux going though the surface dA (ring) is:

$$\frac{\partial Q}{\partial t} = d\xi_e \cdot dA \cdot \cos\theta = d\xi_e \cdot (2\pi \cdot \sin\theta \cdot d\theta) \cdot \cos\theta$$

The power normal to dA and going though dA (ring) is:

$$P_N = \frac{\partial Q_N}{\partial t} = \frac{\partial Q}{\partial t} \cdot \cos\theta$$

Thus:

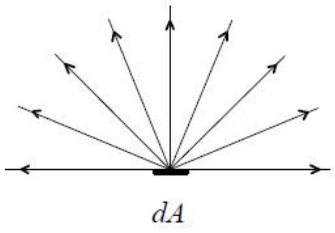
$$P_N = \frac{\partial Q_N}{\partial t} = \frac{\partial Q}{\partial t} \cdot \cos\theta = d\xi_e \cdot (2\pi \cdot \sin\theta \cdot d\theta) \cdot \cos^2\theta$$

Integrating though the complete hemisphere gives:

$$P_N = \frac{\partial Q_N}{\partial t} = \frac{\partial Q}{\partial t} \cdot \cos\theta = \int_0^{\pi/2} d\xi_e \cdot (2\pi \cdot \sin\theta \cdot d\theta) \cdot \cos^2\theta = -\frac{2\pi \cdot \xi_e}{3} \left[\cos^3\theta\right]_0^{\pi/2} = \frac{2\pi \cdot \xi_e}{3} \quad \Rightarrow$$

$$P_N = \frac{2\pi \cdot \xi_e}{3}$$

The same way, we define P_T as the total heat flux going though dA and we write for P_T:



$$P_T = \frac{\partial Q_T}{\partial t} = \int_0^{\pi/2} d\xi_e \cdot (2\pi \cdot \sin\theta \cdot d\theta) \cdot \cos\theta = \xi_e \cdot \pi \cdot \left[\frac{\cos 2\theta}{2}\right]_0^{\pi/2} = \xi_e \cdot \pi \quad \Rightarrow \quad P_N = \frac{2}{3} \cdot P_T$$

Thus, the amount of power per area emitted perpendicularly to the surface of the body is:

$$\boxed{P_N = \frac{2}{3} \cdot \varepsilon\sigma \cdot T^4} \qquad \text{(Equation 30)}$$

1.8 Recoil thermal force on a body of surface Ω and temperature T:

The momentum transferred to the body per unit area of surface per unit of time is:

$$d\left(\frac{\vec{dp}}{dt}\right) = \frac{P_N}{c} \cdot \overrightarrow{dA}$$

Thus, using equation (7), we get:

$$d\left(\frac{\vec{dp}}{dt}\right) = \frac{2\varepsilon\sigma \cdot T^4}{3c} \cdot \overrightarrow{dA}$$

The recoil force df per area of surface dA is:

$$\overrightarrow{df} = -d\left(\frac{\vec{dp}}{dt}\right) = -\frac{2\varepsilon\sigma \cdot T^4}{3c} \cdot \overrightarrow{dA}$$

Finally, the recoil force f on a body of surface Ω is:

$$\boxed{\vec{f} = -\frac{2\varepsilon\sigma}{3c}\int_{\Omega} T^4 \overrightarrow{dA}} \qquad \text{(Equation 31)}$$

1.9 Theory of the penetration depth and phase lag for a thermal wave (general case):

The temperature distribution T(z, t) in a solid material is:

$$\frac{\partial^2 T(z,t)}{\partial z^2} - \frac{\rho C}{K}\frac{\partial T(z,t)}{\partial t} = 0 \qquad (z > 0 \text{ downward})$$

With:

K : Thermal conductivity of the material

C : Specific heat of the material (at constant pressure)

ρ : Density of the material

The boundary conditions at the surface of this solid are:

$$-K\frac{\partial T(z,t)}{\partial z}\Big|_{z=0} = \Re\left[I_o \cdot e^{i\nu t}\right]$$

With:

$I_o \cdot e^{i\nu t}$: Beam of light intensity which is sinusoid variable

ν : Angular intensity variation of the beam of light

We rewrite T(z, t) as a separation of variables between z and t :

T(z, t) = $\Re\left[\Theta(z) \cdot e^{i\nu t}\right]$, we get:

$$\frac{\partial^2 \Theta(z)}{\partial z^2} - \mu^2 \cdot \Theta(z) = 0 \quad \text{With: } \mu = \sqrt{\frac{i\rho C\nu}{K}}$$

Using: $\sqrt{i} = \frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} \cdot i$

We can rewrite μ as:

$$\mu = \sqrt{\frac{i\rho C\nu}{K}} = (1 + i)\sqrt{\frac{\rho C\nu}{K}} = \frac{\sqrt{2}}{2}\left[\frac{1+i}{l_\nu}\right]$$

l_ν is called the penetration depth of the thermal wave and is defined by:

$$\boxed{l_\nu = \sqrt{\frac{K}{\rho C\nu}}} \qquad \text{(Equation 32)}$$

The general solution of the temperature distribution differential equation is then:

$$T(z,t) = \frac{\sqrt{2} \cdot I_o \cdot l_\nu}{2K} \cdot e^{-\frac{z\sqrt{2}}{2 \cdot l_\nu}} \cdot e^{-i\left[\frac{z}{l_\nu} + \frac{\pi}{4}\right]} \cdot \cos(\nu \cdot t)$$

We defined the phase lag Ψ of the thermal wave as:

$$\boxed{\Psi = \frac{z}{l_\nu} + \frac{\pi}{4}} \qquad \text{(Equation 33)}$$

1.10 Recoil thermal force on a body of surface Ω with linearization of T^4:

We already found the recoil force expression on a body of surface Ω in (8) to be:

$$\vec{f} = -\frac{2\varepsilon\sigma}{3c}\int_\Omega T^4 \overrightarrow{dA}$$

We pose: $T = T_o + \Delta T$ and we consider $\frac{\Delta T}{To}$ << 1, we can then make a Taylor development and rewrite T^4 as:

$$T^4 = \left[T_O + \Delta T\right]^4 = \left[T_o\left(1 + \frac{\Delta T}{T_o}\right)\right]^4 \approx T_o^4 + 4T_o^3\Delta T$$

Since: $\overrightarrow{df} = -d\left(\frac{d\vec{p}}{dt}\right)$ and T_o is a constant, only the term $4\cdot T_o^3\Delta T$ remains and we can rewrite $\vec{f}$ as:

$$\boxed{\vec{f} = -\frac{8\varepsilon\sigma}{3c}\cdot T_o^3 \int_\Omega \Delta T\overrightarrow{dA}} \qquad \text{(Equation 34)}$$

1.11 Evaluation of (To, Eo)

Similarly to $T = T_o + \Delta T$, we pose: $E = E_o + \Delta E$ or:

E	=	E_o	+	ΔE
Incoming solar energy		Energy related to T_o		Energy related to ΔT

We define:

a_o: The distance Sun - Earth.

a : The semi-major axis of the asteroid.

ξ_o: The average incoming solar flux density on Earth (ξ_o = 1378 $W\cdot m^{-2}$).

ξ : The average incoming solar flux density on an asteroid located at a distance a from the Sun.

e : The eccentricity of the asteroid.

The energy of the Sun spreads over a sphere of area $4\pi \cdot d^2$. d being the distance between this sphere and the Sun.

Thus, we can write the following relation:

$$4\pi \cdot a_o^2 \cdot \xi_o = 4\pi \cdot a^2 \cdot \xi \qquad \Rightarrow \qquad \xi = \frac{4\pi \cdot a_o^2}{4\pi \cdot a^2} \cdot \xi_o \qquad \Rightarrow \qquad \xi = \left[\frac{a_o}{a}\right]^2 \cdot \xi_o$$

Furthermore, if we consider the body having the shape of a sphere of radius R, the incoming solar radiation is received as a disk of radius R and of area $\pi \cdot R^2$ spreading over the surface of the rotating body of area $4\pi \cdot R^2$.

Thus, the average incident solar radiation ratio on the body's surface will be:

$$\frac{\pi \cdot R^2}{4\pi \cdot R^2} = \frac{1}{4}$$

Finally, the incoming solar energy will also depend of the eccentricity of the orbit of the body with the factor: $\frac{1}{\sqrt{1-e^2}}$. Thus, the Energy E_o can be written as:

$$\boxed{E_o = \frac{\xi_o}{4} \cdot \left[\frac{a_o}{a}\right]^2 \cdot \frac{1}{\sqrt{1-e^2}}} \qquad \text{(Equation 35)}$$

The heat diffusion equation in the volume of a body is (See above 1.9):

$$\vec{\nabla} \cdot \left[K \cdot \vec{\nabla} T\right] = \rho C \frac{\partial T}{\partial t}$$

K : Thermal conductivity of the asteroid

C : Specific heat of the asteroid

ρ : Density of the asteroid

Using again: $T = T_o + \Delta T$, we consider that T varies though time and depends on z (thermal skin depth, z > 0 downward), thus:

$$K\frac{\partial^2 T}{\partial z^2} = \rho C\frac{\partial T}{\partial t}$$

Following the power radiated of a real body expression written in 1.4, the boundary conditions at the surface of the asteroid can be written as:

$$\underbrace{-K\left.\frac{\partial T}{\partial t}\right|_{z=0}}_{\text{Energy conducted inside the asteroid}} + \underbrace{\varepsilon\sigma T^4}_{\text{Incoming energy from the Sun and radiated away}} = \underbrace{(1-A)E}_{\text{Incoming energy from the the sun not reflected away}}$$

We will consider a separation of variables, treating [T_o, E_o] first and [ΔT, E] next.

Thus, the boundary conditions for [T_o, E_o] can be written as:

$$-K\left.\frac{\partial T_o}{\partial t}\right|_{z=0} + \varepsilon\sigma T_o^4 = (1-A)E_o$$

T_o being a constant $\Rightarrow$ $\quad -K\left.\frac{\partial T_o}{\partial t}\right|_{z=0} = 0$

Thus: $\varepsilon\sigma T_o^4 = (1-A)E_o$

Since: $E_o = \frac{\xi_o}{4}\cdot\left[\frac{a_o}{a}\right]^2\cdot\frac{1}{\sqrt{1-e^2}}$

We obtain for T_o : $\boxed{T_o = \left[\left[\frac{a_o}{a}\right]^2\cdot\left[\frac{(1-A)\xi_o}{4\varepsilon\sigma\sqrt{1-e^2}}\right]\right]^{1/4}}$ (Equation 36)

Same result published by Rubincam "Yarkovsky thermal drag on small asteroids" page 1728 [19].

1.12 Evaluation of (ΔT, E)

Using $T^4 \approx T_o^4 + 4 \cdot T_o^4 \Delta T$ and similarly to $[T_o, E_o]$, we write the boundary conditions for $[\Delta T, E]$:

$$- K \left. \frac{\partial \Delta T}{\partial t} \right|_{z=0} + \left. 4\varepsilon\sigma T_o^3 \Delta T \right|_{z=0} = (1 - A)E$$

We now rewrite ΔT and E as a sum of Fourier terms:

$$\Delta T = \Delta T_o \cdot \sum_{p=1}^{\infty} e^{-\mu z} \cdot e^{-ip\nu t}$$

And:

$$E = E_o \cdot \sum_{p=1}^{\infty} e^{-ip\nu t}$$

With:

$$\mu = \frac{\sqrt{2}}{2}\left[\frac{1+i}{l_\nu}\right]$$

ν : Angular intensity variation of incoming solar radiation

l_ν : Penetration depth of the thermal wave

For simplification of the calculation, if we only consider the first term (p = 1) of the Fourier series, we can then write:

For p =1 $\Rightarrow \Delta T = \Delta T_o \cdot e^{-\mu \cdot z} \cdot e^{-ivt}$ And: $E = E_o \cdot e^{-ivt}$

Thus, the boundary conditions for $(\Delta T, E)$ can be rewritten as:

$$-K\frac{\partial(\Delta T_o \cdot e^{-\mu \cdot z} \cdot e^{-ivt})}{\partial t}\bigg|_{z=0} + \varepsilon\sigma T_o^3 \Delta T_o \cdot e^{-\mu \cdot z} \cdot e^{-ivt}\big|_{z=0} = (1-A)E_o \cdot e^{-ivt}$$

But:

$$-K\frac{\partial(\Delta T_o \cdot e^{-\mu \cdot z} \cdot e^{-iv \cdot t})}{\partial t}\bigg|_{z=0} = K\mu \cdot \Delta T_o \cdot e^{-ivt}$$

Thus:

$$K\mu \cdot \Delta T_o \cdot e^{-ivt} + \varepsilon\sigma T_o^3 \Delta T_o \cdot e^{-ivt} = (1-A)E_o \cdot e^{-ivt}$$

Or: $K\mu \cdot \Delta T_o + \varepsilon\sigma T_o^3 \Delta T_o = (1-A)E_o$

And: $\Delta T_o = \frac{(1-A)E_o}{4\varepsilon\sigma T_o^3 + K\mu}$

Since

$\Delta T = \Delta T_o \cdot e^{-\mu \cdot z} \cdot e^{-iv \cdot t}$ and $E = E_o \cdot e^{-iv \cdot t}$

We obtain:

$$E = \frac{1}{4} \cdot \left[\frac{a_o}{a}\right]^2 \cdot \frac{\xi_o}{\sqrt{1-e^2}} \cdot e^{-ivt}$$ (Equation 37)

And:

$$\Delta T = \frac{(1-A)E_o}{4\varepsilon\sigma T_o^3 + K\mu} \cdot e^{-\mu \cdot z} \cdot e^{-ivt}$$ (Equation 38)

Same as result published by Rubincam, David Parry: "Asteroid orbit evolution due to thermal drag" page 1587 [20].

1.13 Evaluation of ΔT using the thermal function Θ1

We try to rewrite $\Delta T_o = \frac{(1-A)E_o}{4\varepsilon\sigma T_o^3 + K\mu}$ as: $\Delta T_o = \frac{B}{1+\chi}$

We pose $\Delta T_o = \frac{B}{1+\chi}$ in order to identify a dimensionless parameter χ

Using: $\mu = \frac{\sqrt{2}}{2}\left[\frac{1+i}{l_v}\right]$

$$\Delta T_o = \frac{(1-A)E_o}{4\varepsilon\sigma T_o^3 + K\mu} = \frac{\frac{(1-A)E_o}{4\varepsilon\sigma T_o^3}}{1+\frac{K\mu}{4\varepsilon\sigma T_o^3}} = \frac{\frac{(1-A)E_o}{4\varepsilon\sigma T_o^3}}{1+\frac{K\sqrt{2}}{2\cdot 4\varepsilon\sigma T_o^3}\left[\frac{1+i}{l_v}\right]}$$

Then using: $l_v = \sqrt{\frac{K}{\rho C v}}$, we get: $\Delta T_o = \frac{\frac{(1-A)E_o}{4\varepsilon\sigma T_o^3}}{1+\frac{\sqrt{K\rho v C}}{4\sqrt{2}\varepsilon\sigma T_o^3}[1+i]}$

Posing: $B = \frac{(1-A)\cdot E_o}{4\varepsilon\sigma T_o^3}$ and: $\Theta_1 = \frac{\sqrt{K\rho v C}}{4\sqrt{2}\varepsilon\sigma T_o^3} = \frac{K}{4\sqrt{2}\cdot\varepsilon\sigma\cdot l_v\cdot T_o^3}$

We can then write:

$$\Delta T_o = \frac{B}{1+\Theta_1[1+i]} = \frac{B}{[1+\Theta_1+i\cdot\Theta_1]} = \frac{B[1+\Theta_1-i\cdot\Theta_1]}{[1+\Theta_1+i\cdot\Theta_1]\cdot[1+\Theta_1-i\cdot\Theta_1]} = \frac{B[1+\Theta_1-i\cdot\Theta_1]}{[1+\Theta_1]^2+\Theta_1{}^2}$$

Or:

$$\Delta T_o = \frac{B\cdot[1+\Theta_1]}{1+2\Theta_1+2\Theta_1{}^2} - \frac{B\cdot\Theta_1}{1+2\Theta_1+2\Theta_1{}^2}i$$

Posing a = $1+\Theta_1$ and b = Θ_1 we can also write:

$a + bi = re^{i\delta} = \sqrt{a^2+b^2}\cdot e^{i\delta} = \sqrt{a^2+b^2}[\cos\delta + i\cdot\sin\delta]$

And: $\tan\delta = \frac{\sin\delta}{\cos\delta} = \frac{a}{b} = \frac{\Theta_1}{\Theta_1 + 1}$

Thus:

$$\sin\delta = \frac{\Theta_1}{\sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}}} \qquad \text{and:} \qquad \cos\delta = \frac{\Theta_1 + 1}{\sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}}}$$

And:

$$e^{-i\delta} = \cos\delta - i \cdot \sin\delta = \frac{\left[1 + \Theta_1 - i \cdot \Theta_1\right]}{\sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}}}$$

Thus:

$$\left[1 + \Theta_1 - i \cdot \Theta_1\right] = \sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}} \cdot \left[\cos\delta - i \cdot \sin\delta\right] = \sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}} \cdot e^{-i\delta}$$

$$\Delta T_o = \frac{B\left[1 + \Theta_1 - i \cdot \Theta_1\right]}{\left[1 + \Theta_1\right]^2 + \Theta_1^{\,2}} = \frac{B\left[1 + \Theta_1 - i \cdot \Theta_1\right]}{\sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}} \cdot \sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}}} = \frac{B \cdot e^{-i\delta}}{\sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}}}$$

Using: $B = \frac{(1 - A)E_o}{4\varepsilon\sigma T_o^3}$ we obtain:

$$\Delta T_o = \frac{(1 - A)E_o e^{-i\delta}}{4\varepsilon\sigma T_o^3 \sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}}}$$

And since: $\Delta T = \Delta T_o \cdot e^{-\mu \cdot z} \cdot e^{-ivt}$, we can write:

$$\Delta T = \frac{(1 - A)E_o e^{-i\delta} \cdot e^{-\mu \cdot z} \cdot e^{-ivt}}{4\varepsilon\sigma T_o^3 \sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}}}$$

Finally, rewriting the boundary conditions at z =0, ΔT becomes:

$$\boxed{\Delta T = \frac{(1 - A)E_o e^{-i\delta} \cdot e^{-ivt}}{4\varepsilon\sigma T_o^3 \sqrt{1 + 2\Theta_1 + 2\Theta_1^{\,2}}}} \qquad \text{(Equation 39)}$$

1.14 Evaluation of the semi-axis acceleration

The acceleration produced by the recoil thermal force defined in (33) is:

$$\frac{\partial^2 r}{\partial t^2} = \frac{f}{m} = -\frac{8\varepsilon\sigma}{3mc} \cdot T_o^3 \int_\Omega \Delta T \overrightarrow{dA}$$

When evaluating ΔT, we only considered the first order p=1 (See 1.12). Thus, we can take the ΔT expression out of the integral:

$$\frac{\partial^2 r}{\partial t^2} = \frac{f}{m} = -\frac{8\varepsilon\sigma}{3mc} \cdot T_o^3 \cdot \Delta T \int_\Omega \overrightarrow{dA}$$

Integrating over an asteroid with a spherical shape and taking only 1/2 of the surface (hemisphere), we get:

$$\frac{\partial^2 r}{\partial t^2} = -\frac{8\varepsilon\sigma}{3mc} \cdot T_o^3 \cdot \Delta T \int_\Omega \overrightarrow{dA} = -\frac{8\varepsilon\sigma}{3mc} \cdot T_o^3 \cdot \Delta T \int R^2 \sin\varphi \cdot d\varphi \cdot d\lambda$$

$$\frac{\partial^2 r}{\partial t^2} = -\frac{8\varepsilon\sigma}{3mc} \cdot T_o^3 \cdot \Delta T \cdot R^2 \cdot \int_0^{\pi/2} \sin\varphi \cdot d\varphi \cdot \int_0^{2\pi} d\lambda$$

$= 2\pi R^2$ (1/2 sphere area)

$$\frac{\partial^2 r}{\partial t^2} = -\frac{16}{3} \cdot \frac{\varepsilon\sigma\pi R^2}{mc} \cdot T_o^3 \cdot \Delta T$$

Using: $m = \frac{4}{3} \cdot \pi R^3 \rho \quad \Rightarrow \quad \frac{1}{m} = \frac{3}{4\pi\rho R^3}$

$$\frac{\partial^2 r}{\partial t^2} = -\frac{4\varepsilon\sigma T_o^3}{\rho R c} \cdot \Delta T$$

Using the result from (38), $\Delta T = \frac{(1-A)E_o e^{-i\delta}\cdot e^{-i\nu t}}{4\varepsilon\sigma T_o^3\sqrt{1+2\Theta_1+2\Theta_1{}^2}}$

$$\frac{\partial^2 r}{\partial t^2} = -\frac{4\varepsilon\sigma T_o^3}{\rho Rc}\cdot\frac{(1-A)E_o e^{-i\delta}\cdot e^{-i\nu t}}{4\varepsilon\sigma T_o^3\sqrt{1+2\Theta_1+2\Theta_1{}^2}} = -\frac{(1-A)E_o e^{-i\delta}\cdot e^{-i\nu t}}{\rho Rc\sqrt{1+2\Theta_1+2\Theta_1{}^2}}$$

Thus:

$$\boxed{\frac{\partial^2 r}{\partial t^2} = -\frac{(1-A)E_o e^{-i\delta}\cdot e^{-i\nu t}}{\rho Rc\sqrt{1+2\Theta_1+2\Theta_1{}^2}}}$$ (Equation 40)

With E_o already calculated in (34): $\boxed{E_o = \frac{\xi_o}{4}\cdot\left[\frac{a_o}{a}\right]^2\cdot\frac{1}{\sqrt{1-e^2}}}$

1.15 Evaluation of the mean semi-axis drift or mean motion

The mean motion along the spin axis z of the asteroid (also called seasonal Yarkosvsky effect) is defined as:

$\left[\frac{da}{dt}\right]_S = \frac{2}{n}\cdot P\left[\frac{\partial^2 r}{\partial t^2}\right]$ = Projection of the acceleration component on the spin z axis of the studied asteroid.

In this case, n the mean motion, is equal to ν the angular intensity variation of the incoming solar radiation

We also need to average all orientations by an extra factor of $\frac{2}{3}$.

See Rubincam "Yarkovsky thermal drag on small asteroids" page 1726 [19].

Thus:

$$\left[\frac{da}{dt}\right]_S = \frac{2}{n}\cdot\frac{2}{3}\cdot P\left[\frac{\partial^2 r}{\partial t^2}\right] = \frac{4}{3n}\cdot P\left[\frac{\partial^2 r}{\partial t^2}\right]$$ P is the projection on the axis z

We then consider the photocentric frame, using $\hat{s} = \hat{s}_1 + i\cdot\hat{s}_2$

See Vokroukicky: " Thermal forces on planetary ring particles: application to the main system of Saturn" page 721 [21].

$\hat{s}_1$ is the projection on the spin vector $\hat{s}$.

$\hat{s}_2$ is the projection on the plan x,y of the equatorial plan.

This implies $s_1 = 0$ and $s_2 = \sin\gamma$.

Since γ is the obliquity between the spin axis z and the normal to the equatorial plan. $\Rightarrow$ $\hat{s} = i\cdot\sin\gamma$

As a result, we will only consider the projection of $\frac{\partial^2 r}{\partial t^2}$ on the z axis and write:

$$\left[\frac{da}{dt}\right]_S = \frac{4}{3n}\cdot\Im\left[\frac{\partial^2 r}{\partial t^2}\right]$$ $\Im$ is the imaginary part of the complex

acceleration $\frac{\partial^2 r}{\partial t^2}$ found in (14).

From equation: $\overrightarrow{df} = \frac{\xi}{c}\cdot\hat{s}\cdot\hat{s}\cdot\hat{n}\cdot dA$ defined in (2) and using: $\hat{s} = i\cdot\sin\gamma$, we can then write:

$$\left[\frac{da}{dt}\right]_S = \frac{4}{3n}\cdot \Im\left[\frac{\partial^2 r}{\partial t^2}\right]\cdot \hat{s}\cdot \hat{s} = \frac{4}{3n}\cdot \Im\left[\frac{\partial^2 r}{\partial t^2}\right]\cdot \sin^2\gamma$$

Using the previous result: $\frac{\partial^2 r}{\partial t^2} = -\frac{(1-A)\cdot E_o e^{-i\delta}}{\rho Rc\sqrt{1+2\Theta_1+2\Theta_1^2}}$ in (14) and choosing t = 0

since we already averaged out with the mean motion, we obtain:

$$\left[\frac{da}{dt}\right]_S = \frac{4}{3n}\cdot \Im\left[-\frac{(1-A)\cdot E_o e^{-i\delta}}{\rho Rc\sqrt{1+2\Theta_1+2\Theta_1^2}}\right]\cdot \sin^2\gamma = \left[-\frac{4(1-A)\cdot E_o}{3n\rho Rc\sqrt{1+2\Theta_1+2\Theta_1^2}}\right]\cdot \Im\left[e^{-i\delta}\right]\cdot \sin^2\gamma$$

$$\Im\left[e^{-i\delta}\right] = \Im\left[\cos\delta - i\cdot\sin\delta\right] = -\sin\delta$$

Using: $\sin\delta = \frac{\Theta_1}{\sqrt{1+2\Theta_1+2\Theta_1^2}}$

We obtain: $\Im\left[e^{-i\delta}\right] = -\frac{\Theta_1}{\sqrt{1+2\Theta_1+2\Theta_1^2}}$

Thus:

$$\left[\frac{da}{dt}\right]_S = \frac{4(1-A)\cdot E_o}{3n\rho Rc\sqrt{1+2\Theta_1+2\Theta_1^2}}\cdot\frac{\Theta_1}{\sqrt{1+2\Theta_1+2\Theta_1^2}}\cdot\sin^2\gamma = \frac{4(1-A)\cdot E_o\cdot\Theta_1}{3n\rho Rc\left[1+2\Theta_1+2\Theta_1^2\right]}\cdot\sin^2\gamma$$

Finally, we use the expression of E_o found in (11a). We had:

$$E_o = \frac{\xi_o}{4}\left[\frac{a_o}{a}\right]^2\cdot\frac{1}{\sqrt{1-e^2}}$$

If we choose: $E_o = \frac{\xi_o}{4}\left[\frac{a_o}{a}\right]^2$ for a circular orbit approximation (eccentricity e=0).

We obtain:

$$\left[\frac{da}{dt}\right]_S = \frac{(1-A)\cdot a_o^2\cdot\xi_o\cdot\Theta_1}{3a^2\rho nRc\left[1+2\Theta_1+2{\Theta_1}^2\right]}\cdot\frac{1}{\sqrt{1-e^2}}\cdot\sin^2\gamma \qquad \text{(Equation 41)}$$

Same result published by E.Skojlöv: "The influence of the spin vectors of asteroids from the Yarkovsky effect" [22].

We define the function: $F_{n,1} = \left[\frac{a_o}{a}\right]^2\cdot\left[\frac{\Theta_1}{1+2\Theta_1+2{\Theta_1}^2}\right]\cdot\frac{1}{\sqrt{1-e^2}}$, we can rewrite $\left[\frac{da}{dt}\right]_S$ as:

$$\left[\frac{da}{dt}\right]_S = \frac{\alpha\cdot\xi_o\cdot F_{n,1}\cdot\sin^2\gamma}{3\rho nRc} \qquad \text{(Equation 42)}$$

With: $F_{n,1} = \left[\frac{a_o}{a}\right]^2\cdot\left[\frac{\Theta_1}{1+2\Theta_1+2{\Theta_1}^2}\right]\cdot\frac{1}{\sqrt{1-e^2}}$ $\qquad \Theta_1 = \frac{K}{4\sqrt{2}\varepsilon\sigma\cdot l_n\cdot T_o^3}$ and: $l_n = \sqrt{\frac{K}{\rho Cn}}$

Using again: $m = \frac{4}{3}\cdot\pi R^3\rho \quad \Rightarrow \quad \frac{1}{\rho R} = \frac{4\pi R^2}{3m}$

We get: $\left[\frac{da}{dt}\right]_S = \frac{4\pi\alpha R^2\cdot\xi_o\cdot F_{n,1}\cdot\sin^2\gamma}{9nc} = \frac{4\pi\alpha R^2\cdot\xi_o\cdot F_{n,1}\cdot\sin^2\gamma}{9nc}$

Posing: $\Phi = \frac{\pi R^2\xi_o}{mc}$ we obtain:

$$\left[\frac{da}{dt}\right]_S = \frac{4\alpha}{9}\cdot\frac{\Phi}{n}\cdot F_{n,1}\cdot\sin^2\gamma \qquad \text{(Equation 43)}$$

Same result as the seasonal expression (21) we introduced in this annexe. (With p=1 since we only considered the first order).

Note: With a further expansion of the Fourier expressions with terms p > 1, the function $F_{n,1}$ would have become: $F_{n,p} = \left[\frac{a_o}{a}\right]^2 \cdot \left[\frac{\kappa_1(R')\Theta_1}{1 + 2\kappa_2(R')\Theta_1 + \kappa_3(R')\Theta_1{}^2}\right] \cdot \frac{1}{\sqrt{1-e^2}}$

See Bottke, W. F., Jr.; Vokrouhlický, D.; Rubincam, D. P.; Broz, M.: "The Effect of Yarkovsky Thermal Forces on the Dynamical Evolution of Asteroids and Meteoroids" page 399 [23].

APPENDIX B: ACCESS QUERIES

The Access query "*qry_sup_year*" selects all the 44,562 picture rows from the Subaru header file, calculates the equatorial coordinates (λ,β) of each picture, calculates the decimal RA and DEC coordinates of the center of each picture and inserts them in the "tbl_frames_gen" table.

```
SELECT Year([FILTER0]) AS [Year], SUP_YEARS.[#FRAME_ID],
Val([EXPTIME]) AS expo_duration, SUP_YEARS.FILTER0,
SUP_YEARS.FILTER, SUP_YEARS.RA2000, SUP_YEARS.DEC2000,
Round(Left([DEC2000],1) &
Right(Left([DEC2000],3),2)+Left(Right([DEC2000],8),2)/60+Right([DEC2000],5)/3
600,4) AS DEC_DECI,
Round((Left([RA2000],2)+Right(Left([RA2000],5),2)/60+Right([RA2000],6)/3600)*
15,4) AS RA_DECI, Sin(Deg2Rad([DEC_DECI])) AS SIN_DELTA,
Cos(Deg2Rad([DEC_DECI])) AS COS_DELTA, Cos(Deg2Rad([RA_DECI])) AS
COS_ALPHA, Sin(Deg2Rad([RA_DECI])) AS SIN_ALPHA, 23.441884 AS
EPSILON, Cos(Deg2Rad([EPSILON])) AS COS_EPSILON,
Sin(Deg2Rad([EPSILON])) AS SIN_EPSILON, [SIN_DELTA]*[COS_EPSILON]-
[COS_DELTA]*[SIN_EPSILON]*[SIN_ALPHA] AS SIN_BETA,
Tan(Deg2Rad([DEC_DECI])) AS TAN_DELTA, 180/PI()*ArcSin([SIN_BETA]) AS
BETA, [SIN_ALPHA]*[COS_EPSILON]+[TAN_DELTA]*[SIN_EPSILON] AS
YVAL, [COS_ALPHA] AS XVAL,
IIf(180/PI()*ATan2([COS_ALPHA],[YVAL])<0,180/PI()*ATan2([COS_ALPHA],[YV
AL])+360,180/PI()*ATan2([COS_ALPHA],[YVAL])) AS LAMBDA,
Val(Left([RA2000],2)) AS RACALC, Val(Left([DEC2000],3)) AS DECCALC,
SUP_YEARS.UT_STR, SUP_YEARS.EXPTIME, SUP_YEARS.ALTITUDE,
SUP_YEARS.DATA_TYP, SUP_YEARS.OBJECT, SUP_YEARS.PSF_ELLIP,
SUP_YEARS.SKY_LEVEL INTO TBL_FRAMES_GEN
FROM SUP_YEARS
ORDER BY Year([FILTER0]) DESC , SUP_YEARS.FILTER DESC ,
SUP_YEARS.RA2000;
```

The Access query "qry_sup_years_filter" selected 2,078 pictures out of the

44,562 available pictures from the "tbl_frames_gen" table.

These 2.078 pictures are selected from the criteria below and are then inserted in

the MySQL "tbl_picture" table.

Criteria used:

1- Pictures whose ecliptic latitude is within ± 30^{o} from the ecliptic.
2- Pictures that are "non-calibration frame“.
3- Pictures that have an exposure time of at least 100 seconds.
4- Pictures that have the most sensitive filters, red and infrared (~0.7 μm).

Access query "QRY _SUP_YEARS_FILTER" that filters the table

TBL_FRAMES_GEN:

```
SELECT TBL_FRAMES_GEN.Year, TBL_FRAMES_GEN.[#FRAME_ID],
TBL_FRAMES_GEN.OBJECT, TBL_FRAMES_GEN.expo_duration,
TBL_FRAMES_GEN.FILTER, TBL_FRAMES_GEN.FILTER0,
TBL_FRAMES_GEN.RA2000, TBL_FRAMES_GEN.RACALC,
TBL_FRAMES_GEN.DEC2000, TBL_FRAMES_GEN.DECCALC,
TBL_FRAMES_GEN.BETA, TBL_FRAMES_GEN.LAMBDA,
TBL_FRAMES_GEN.UT_STR, TBL_FRAMES_GEN.EXPTIME,
TBL_FRAMES_GEN.ALTITUDE, TBL_FRAMES_GEN.DATA_TYP,
TBL_FRAMES_GEN.PSF_ELLIP, TBL_FRAMES_GEN.SKY_LEVEL
FROM TBL_FRAMES_GEN
WHERE (((TBL_FRAMES_GEN.expo_duration)>=100) AND
((TBL_FRAMES_GEN.FILTER)="W-C-RC") AND
((TBL_FRAMES_GEN.BETA)>=-30 And (TBL_FRAMES_GEN.BETA)<=30) AND
((TBL_FRAMES_GEN.LAMBDA)>=0 And
(TBL_FRAMES_GEN.LAMBDA)<=360) AND
((TBL_FRAMES_GEN.DATA_TYP)="OBJECT")) OR
(((TBL_FRAMES_GEN.expo_duration)>=100) AND
((TBL_FRAMES_GEN.FILTER)="W-C-IC") AND
((TBL_FRAMES_GEN.BETA)>=-30 And (TBL_FRAMES_GEN.BETA)<=30) AND
((TBL_FRAMES_GEN.LAMBDA)>=0 And
(TBL_FRAMES_GEN.LAMBDA)<=360) AND
((TBL_FRAMES_GEN.DATA_TYP)="OBJECT"));
```

The Access query "qry_populate_picture_local_access" updates the RA_MIN_CALC and RA_MAX_CALC of each picture row in the tbl_pictures table by adding or subtracting half the height of the picture (± 0.355).

It also updates the DEC_MIN_CALC and DEC_MAX_CALC by adding or subtracting half the width of the picture (± 0.2319).

```
INSERT INTO tbl_pictures2 ( p1, FRAME_ID, DATE_OBS, FILTER, RA2000,
DEC2000, UT_STR, EXPTIME, ALTITUDE, DATA_TYP, [OBJECT], PSF_ELLIP,
SKY_LEVEL, RA_MIN_CALC, RA_MAX_CALC, DEC_MIN_CALC,
DEC_MAX_CALC )
SELECT tbl_picture.p1, [tbl_picture].[#FRAME_ID], tbl_picture.[Date Obs],
tbl_picture.FILTER, tbl_picture.RA2000, tbl_picture.DEC2000,
tbl_picture.UT_STR, tbl_picture.expo_duration, tbl_picture.ALTITUDE,
tbl_picture.DATA_TYP, tbl_picture.OBJECT, tbl_picture.PSF_ELLIP,
tbl_picture.SKY_LEVEL, [tbl_picture].[RACALC]-0.355 AS RA_MIN_RECALC,
[tbl_picture].[RACALC]+0.355 AS RA_MAX_RECALC, [tbl_picture].[DECCALC]-
0.2319 AS DEC_MIN_RECALC, [tbl_picture].[DECCALC]+0.232 AS
DEC_MAX_RECALC
FROM tbl_picture;
```

The Access query "qry_asteroids_to_keep" filters the tbl_matches table by comparing the decimal values of the RA and DEC calculated by the Horizons online system with the min, max of RA and the min and max of DEC which represent the external boundaries of the Subaru CCD picture. Any asteroid whose calculated Horizons RA and DEC are found within the boundaries of the CCD pictures are selected and saved in an Excel spreadsheet for future visual confirmation.

```
SELECT tbl_pictures2.p1, tbl_matches.PIC_FRAME_ID, tbl_pictures2.OBJECT,
tbl_matches.AST_id, tbl_matches.AST_name, tbl_matches.PIC_DATE_OBS,
tbl_pictures2.RA2000, tbl_pictures2.DEC2000, tbl_pictures2.RA_MIN_CALC,
tbl_pictures2.RA_MAX_CALC, tbl_pictures2.DEC_MIN_CALC,
tbl_pictures2.DEC_MAX_CALC, tbl_matches.AST_DEC_CALC,
tbl_matches.AST_RA_CALC, tbl_matches.HORIZONS_LINE,
tbl_matches.PIC_UT_STR, tbl_matches.HORIZONS_RA,
tbl_matches.HORIZONS_DEC, tbl_matches.HORIZONS_RA_val,
tbl_matches.HORIZONS_DEC_val, tbl_matches.MAGNITUDE
FROM tbl_pictures2 INNER JOIN tbl_matches ON tbl_pictures2.p1 =
tbl_matches.PIC_frame_pk
WHERE
(((tbl_matches.HORIZONS_DEC_val)>=[tbl_pictures2].[DEC_MIN_CALC] And
(tbl_matches.HORIZONS_DEC_val)<=[tbl_pictures2].[DEC_MAX_CALC]) AND
((tbl_matches.HORIZONS_RA_val)>=[tbl_pictures2].[RA_MIN_CALC] And
(tbl_matches.HORIZONS_RA_val)<=[tbl_pictures2].[RA_MAX_CALC]) AND
((tbl_pictures2.RA_MIN_CALC) Is Not Null))
GROUP BY tbl_pictures2.p1, tbl_matches.PIC_FRAME_ID,
tbl_pictures2.OBJECT, tbl_matches.AST_id, tbl_matches.AST_name,
tbl_matches.PIC_DATE_OBS, tbl_pictures2.RA2000, tbl_pictures2.DEC2000,
tbl_pictures2.RA_MIN_CALC, tbl_pictures2.RA_MAX_CALC,
tbl_pictures2.DEC_MIN_CALC, tbl_pictures2.DEC_MAX_CALC,
tbl_matches.AST_DEC_CALC, tbl_matches.AST_RA_CALC,
tbl_matches.HORIZONS_LINE, tbl_matches.PIC_UT_STR,
tbl_matches.HORIZONS_RA, tbl_matches.HORIZONS_DEC,
tbl_matches.HORIZONS_RA_val, tbl_matches.HORIZONS_DEC_val,
tbl_matches.MAGNITUDE ORDER BY tbl_pictures2.p1;
```

APPENDIX C: STORED PROCEDURES AND PROGRAMS

The stored procedure "pic_FIND_MATCH" calculates the decimal RA and DEC positions of each asteroid and compares them with the external boundary of a circle of radius 1.4 arc seconds on each picture. All the asteroids found inside this circle in each picture are inserted in the "tbl_matches" table. The method used for the calculation of RA and DEC follows the instruction given by [24].

```
DELIMITER $$

DROP PROCEDURE IF EXISTS `pic_FIND_MATCH15` $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `pic_FIND_MATCH15`()
BEGIN
DECLARE v_p1,v_f1,b,c INT;
DECLARE l_department_id INT;
DECLARE l_employee_id   INT;
DECLARE l_emp_count     INT DEFAULT 0 ;
DECLARE l_done          INT DEFAULT  0;

DECLARE v_DATE_OBS DATE;
DECLARE v_UT_STR VARCHAR(50);
DECLARE v_FRAME_ID VARCHAR(50);
DECLARE v_RA_CALC REAL(15,12);
DECLARE v_DEC_CALC REAL(15,12);
DECLARE v_year_pic INT;
DECLARE v_month_pic INT;
DECLARE v_day_pic INT;
DECLARE v_UT_STR_HRS INT;
DECLARE v_UT_STR_MIN INT;
DECLARE v_UT_STR_SEC INT;

DECLARE v_f2  varchar(50);
DECLARE v_f4  float(9,6);
DECLARE v_f12 date;
DECLARE v_f13 float(9,6);
DECLARE v_f14 float(9,6);
DECLARE v_f15 float(9,6);
DECLARE v_f16 float(9,6);
DECLARE v_f17 float(10,8);
DECLARE v_f18 real(25,10);
```

```
DECLARE v_pi REAL;
DECLARE v_asteroidid INT;
DECLARE v_asteroidname varchar(50);
DECLARE v_epoch_osculation date;
DECLARE v_mean_anomaly1 float(9,6);
DECLARE v_arg_perihelion float(9,6);
DECLARE v_long_asc_node float(9,6);
DECLARE v_inclin_of_orbit float(9,6);
DECLARE v_inclin_orbit float(9,6);
DECLARE v_excentricity float(10,8);
DECLARE v_mean_distance real(25,10);
DECLARE v_abs_magnitude float(10,8);
DECLARE v_magnitude float(10,8);

DECLARE v_year_eo INT;
DECLARE v_month_eo INT;
DECLARE v_day_eo INT;
DECLARE v_ut2 real;
DECLARE v_R  real;
DECLARE v_P  real;
DECLARE v_alpha1  real;
DECLARE v_ut real;
DECLARE v_julian_days real;
DECLARE v_DT real;
DECLARE v_M2 real;
DECLARE v_mean_anomaly2 real;
DECLARE v_A real;
DECLARE v_long_perihelion real;
DECLARE v_E_tolerance real;
DECLARE v_excentric_anomaly real;
DECLARE v_E real;
DECLARE v_ex real;
DECLARE v_t real;
DECLARE v_true_anomaly real;
DECLARE v_l real;
DECLARE v_heliocentric_long real;
DECLARE v_heliocentric_dist real;
DECLARE v_phi real;
DECLARE v_heliocentric_lat real;
DECLARE v_x_cart_ecliptic_coord real;
DECLARE v_y_cart_ecliptic_coord real;
DECLARE v_phi_ecliptic real;
DECLARE v_polar_ecliptic_angle real;
DECLARE v_heliocentric_long_proj real;
```

```
DECLARE v_heliocentric_dist_proj real;
DECLARE v_mean_distance_earth_o real;
DECLARE v_excentricity_earth_o real;
DECLARE v_inclin_orbit_earth_o real;
DECLARE v_long_perihelion_earth_o real;
DECLARE v_long_asc_node_earth_o real;
DECLARE v_mean_long_earth_o real;
DECLARE v_mean_distance_earth_r real;
DECLARE v_excentricity_earth_r real;
DECLARE v_inclin_orbit_earth_r real;
DECLARE v_long_perihelion_earth_r real;
DECLARE v_long_asc_node_earth_r real;
DECLARE v_mean_long_earth_r real;
DECLARE v_mean_long_earth_factor_r real;
DECLARE v_ut_earth real;
DECLARE v_julian_century_time real;
DECLARE v_t_e real;
DECLARE v_mean_distance_earth_t real;
DECLARE v_excentricity_earth_t real;
DECLARE v_long_perihelion_earth_t real;
DECLARE v_long_asc_node_earth_t real;
DECLARE v_mean_long_earth_t real;
DECLARE v_mean_anomaly_earth_t real;
DECLARE v_exc_anomaly_earth_t real;
DECLARE v_true_anomaly_earth_t real;
DECLARE v_heliocentric_dist_earth_t real;
DECLARE v_geo_ecliptic_long real;
DECLARE v_lambda real;
DECLARE v_geo_ecliptic_lat real;
DECLARE v_beta real;
DECLARE v_epsilon_century_time real;
DECLARE v_ut_epsilon real;
DECLARE v_epsilon real;
DECLARE v_ecliptic_mean_obliquity real;
DECLARE v_SIN_DELTA real;
DECLARE v_declination real;
DECLARE v_alpha_y real;
DECLARE v_alpha_x real;
DECLARE v_RA real;
DECLARE v_right_ascention real;
DECLARE v_right_ascention_calc real;
DECLARE v_declination_calc real;
DECLARE v_match1 varchar(1);
DECLARE v_delta_RA real;
```

```
DECLARE v_delta_dec real;
DECLARE v_ecliptic_longitude_PIC real;
DECLARE v_delta_PROB float;

DECLARE cur_1 cursor FOR SELECT
p1,DATE_OBS,UT_STR,FRAME_ID,RA_CALC,DEC_CALC  from tbl_pictures;
DECLARE cur_2 cursor FOR SELECT f1,f2,f4,f12,f13,f14,f15,f16,f17,f18 from
tbl_astorb2  where f1<=5000;

DECLARE CONTINUE HANDLER FOR NOT FOUND SET l_done=1;

OPEN cur_1;
cur_1_loop: LOOP
FETCH cur_1 INTO
v_p1,v_DATE_OBS,v_UT_STR,v_FRAME_ID,v_RA_CALC,v_DEC_CALC;

SET v_year_pic=YEAR(v_DATE_OBS);
SET v_month_pic=MONTH(v_DATE_OBS);
SET v_day_pic=DAY(v_DATE_OBS);
SET v_UT_STR_HRS=LEFT(v_UT_STR,2);
SET v_UT_STR_MIN=LEFT(SUBSTRING(v_UT_STR,4),2);
SET v_UT_STR_SEC=LEFT(SUBSTRING(v_UT_STR,7),6);

IF l_done=1 THEN
LEAVE cur_1_loop;
END IF;

  OPEN cur_2;
  SET l_emp_count=0;
  cur_2_loop: LOOP
  FETCH cur_2 INTO v_f1,v_f2,v_f4,v_f12,v_f13,v_f14,v_f15,v_f16,v_f17,v_f18;

  IF l_done=1 THEN
   LEAVE cur_2_loop;
  END IF;

  SET v_pi=PI();
  SET v_abs_magnitude=v_f4;
  SET v_asteroidid=v_f1;
  SET v_asteroidname=v_f2;
  SET v_epoch_osculation=v_f12;
  SET v_mean_anomaly1=v_f13;
  SET v_arg_perihelion=v_f14;
  SET v_long_asc_node=v_f15;
```

```
  SET v_inclin_of_orbit=v_f16;
  SET v_inclin_orbit=v_f16;
  SET v_excentricity=v_f17;
  SET v_mean_distance=v_f18;
  SET v_abs_magnitude=v_f4;
  SET v_year_eo=YEAR(v_epoch_osculation);
  SET v_month_eo=MONTH(v_epoch_osculation);
  SET v_day_eo=DAY(v_epoch_osculation);

  SET v_ut2=ut_time(v_year_eo,v_month_eo,v_day_eo,0,0,0);

  SET v_R= POW(v_mean_distance, 3)*1.3341260180*POW(10, 5);
  SET v_P=pow(v_R, 0.5);
  SET v_alpha1=360/v_P;
  SET
v_ut=ut_time(v_year_pic,v_month_pic,v_day_pic,v_UT_STR_HRS,v_UT_STR_
MIN,v_UT_STR_SEC);
  SET v_julian_days=v_ut;
  SET v_DT=v_ut-v_ut2;
  SET v_M2 = v_mean_anomaly1 + v_alpha1*(v_ut-v_ut2);
  SET v_mean_anomaly2=rev(v_M2);
  SET v_A=v_arg_perihelion + v_long_asc_node;
  SET v_long_perihelion=rev(v_A);

  SET v_E_tolerance=0.0000001;
  SET
v_excentric_anomaly=ec_anomaly(v_mean_anomaly2,v_excentricity,v_E_tolera
nce);

  SET v_t=f_true_anomaly(v_excentric_anomaly,v_excentricity);
  SET v_true_anomaly=rev(v_t);

  SET v_l=v_true_anomaly+v_long_perihelion;
  SET v_heliocentric_long=rev(v_l);

  SET
v_ut_epsilon=ut_time_epsilon(v_year_pic,v_month_pic,v_day_pic,v_UT_STR_H
RS,v_UT_STR_MIN,v_UT_STR_SEC);
  SET v_epsilon_century_time=v_ut_epsilon/36525;
  SET v_ecliptic_mean_obliquity=23.452294 - (  46.845*v_epsilon_century_time
+ 0.0059*POW(v_epsilon_century_time,2) -
0.00181*POW(v_epsilon_century_time,3)  ) / 3600;
  SET v_epsilon=v_ecliptic_mean_obliquity;
```

```
  SET
v_ecliptic_longitude_PIC=rev(180/v_pi*ATAN2(SIN(RADIANS(v_RA_CALC))*CO
S(RADIANS(v_epsilon))+TAN(RADIANS(v_DEC_CALC))*SIN(RADIANS(v_epsil
on)),COS(RADIANS(v_RA_CALC))));

  IF ABS(v_heliocentric_long-v_ecliptic_longitude_PIC)<=40 THEN

   SET
v_heliocentric_dist=f_heliocentric_dist(v_mean_distance,v_excentricity,v_true_a
nomaly);

   SET
v_phi=f_heliocentric_lat(v_heliocentric_long,v_long_asc_node,v_inclin_orbit);

   SET v_heliocentric_lat=rev(v_phi);

   SET v_x_cart_ecliptic_coord=cos(RADIANS(v_heliocentric_long-
v_long_asc_node));
   SET v_y_cart_ecliptic_coord=sin(RADIANS(v_heliocentric_long-
v_long_asc_node))*cos(RADIANS(v_inclin_orbit));

   SET
v_phi_ecliptic=180/v_pi*ATAN2(v_y_cart_ecliptic_coord,v_x_cart_ecliptic_coord)
;
   SET v_polar_ecliptic_angle=rev(v_phi_ecliptic);

   SET v_heliocentric_long_proj=rev(v_polar_ecliptic_angle + v_long_asc_node);

   SET
v_heliocentric_dist_proj=v_heliocentric_dist*COS(RADIANS(v_heliocentric_lat));


   SET v_mean_distance_earth_o=1.00000011;
   SET v_excentricity_earth_o=0.01671022;
   SET v_inclin_orbit_earth_o=2.77274871;
   SET v_long_perihelion_earth_o=102.94719;
   SET v_long_asc_node_earth_o=348.73936;
   SET v_mean_long_earth_o=100.46435;
   SET v_mean_distance_earth_r=-5/100000000;
   SET v_excentricity_earth_r=-0.00003804;
   SET v_inclin_orbit_earth_r=-46.94;
   SET v_long_perihelion_earth_r=1198.28;
   SET v_long_asc_node_earth_r=-18228.25;
```

```
  SET v_mean_long_earth_r=1293740.63;
  SET v_mean_long_earth_factor_r=99;

  SET
v_ut_earth=ut_time_earth(v_year_pic,v_month_pic,v_day_pic,v_UT_STR_HRS,v
_UT_STR_MIN,v_UT_STR_SEC);
  SET v_julian_century_time=v_ut_earth/36525;
  SET v_t_e=v_julian_century_time;
  SET
v_mean_distance_earth_t=v_mean_distance_earth_o+v_mean_distance_earth_
r*v_t_e;
  SET v_excentricity_earth_t=v_excentricity_earth_o +
v_excentricity_earth_r*v_t_e;
  SET v_long_perihelion_earth_t=v_long_perihelion_earth_o +
(v_long_perihelion_earth_r / 3600)*v_t_e;

  SET v_long_asc_node_earth_t=v_long_asc_node_earth_o +
(v_long_asc_node_earth_r / 3600)*v_t_e;


  SET v_mean_long_earth_t=rev(v_mean_long_earth_o +
(v_mean_long_earth_r / 3600 + 360*v_mean_long_earth_factor_r)*v_t_e);


  SET v_mean_anomaly_earth_t=v_mean_long_earth_t-
v_long_perihelion_earth_t;

  SET
v_exc_anomaly_earth_t=ec_anomaly(v_mean_anomaly_earth_t,v_excentricity_e
arth_t,v_E_tolerance);

  SET
v_true_anomaly_earth_t=f_true_anomaly(v_exc_anomaly_earth_t,v_excentricity
_earth_t);

  SET v_heliocentric_dist_earth_t=(v_mean_distance_earth_t*(1-
v_excentricity_earth_t*v_excentricity_earth_t))/(1+v_excentricity_earth_t*COS(R
ADIANS(v_true_anomaly_earth_t)));

  SET
v_lambda=f_geo_ecliptic_long(v_mean_long_earth_t,v_heliocentric_dist_proj,v_
heliocentric_dist_earth_t,v_heliocentric_long_proj);
  SET v_geo_ecliptic_long=rev(v_lambda);
```

```
  SET
v_beta=f_geo_ecliptic_lat(v_mean_long_earth_t,v_heliocentric_dist_proj,v_helioc
entric_dist_earth_t,v_heliocentric_long_proj,v_heliocentric_lat,v_geo_ecliptic_lon
g);
  SET  v_geo_ecliptic_lat=rev(v_beta);


  SET v_epsilon=v_ecliptic_mean_obliquity;
  SET v_beta=v_geo_ecliptic_lat;
  SET v_lambda=v_geo_ecliptic_long;
  SET
v_SIN_DELTA=SIN(RADIANS(v_beta))*COS(RADIANS(v_epsilon))+COS(RADI
ANS(v_beta))*SIN(RADIANS(v_epsilon))*SIN(RADIANS(v_lambda));
  SET v_declination=180/v_pi*ASIN(v_SIN_DELTA);


  SET v_beta=v_geo_ecliptic_lat;
  SET v_lambda=v_geo_ecliptic_long;
  SET v_alpha_y=SIN(RADIANS(v_lambda))*COS(RADIANS(v_epsilon)) -
TAN(RADIANS(v_beta))*SIN(RADIANS(v_epsilon));
  SET v_alpha_x=COS(RADIANS(v_lambda));
  SET v_RA=180/v_pi*ATAN2(v_alpha_y,v_alpha_x);
  SET v_right_ascention=rev(v_RA);
  SET v_right_ascention_calc=v_right_ascention;
  SET v_declination_calc=v_declination;
  SET v_delta_RA=1;
  SET v_delta_dec=1;
  IF (ABS(v_RA_CALC-v_right_ascention_calc)<=v_delta_RA AND
ABS(v_DEC_CALC-v_declination_calc)<=v_delta_dec) THEN
    SET v_delta_dec= ABS(v_DEC_CALC-v_declination_calc);
    SET v_delta_RA= ABS(v_RA_CALC-v_right_ascention_calc);
    SET v_match1='Y';
    SET
v_magnitude=f_magnitude(v_mean_distance_earth_t,v_heliocentric_dist,v_mean
_long_earth_t,v_heliocentric_long,v_abs_magnitude);
  ELSE
    SET v_match1='N';
  END IF;

IF (v_match1='Y') THEN

    SET v_delta_PROB=ABS(ROUND((1-POW(v_delta_dec*v_delta_dec +
(v_delta_RA*v_delta_RA)*ABS(cos(RADIANS((v_DEC_CALC+v_RA_CALC))/2))
,0.5))*100,2));
```

```
INSERT INTO tbl_matches7
(PIC_frame_pk,PIC_FRAME_ID,PIC_RA_CALC,PIC_DEC_CALC,PIC_DATE_O
BS,PIC_UT_STR,AST_id,AST_name,AST_RA_CALC,AST_DEC_CALC,DELTA
_RA,DELTA_REC,MAGNITUDE,DELTA_PROB) values
(v_p1,v_FRAME_ID,v_RA_CALC,v_DEC_CALC,v_DATE_OBS,v_UT_STR,v_f1,
v_f2,v_right_ascention_calc,v_declination_calc,v_delta_RA,v_delta_dec,v_magni
tude,v_delta_PROB);
   END IF;
 END IF;

END LOOP;
CLOSE cur_2;
SET l_done=0;


END LOOP cur_1_loop;
CLOSE cur_1;

END $$

DELIMITER ;
```

Function "ut_time_earth()":

This function calculates the Earth UT time from the date and UT time a picture

was taken.

```
DELIMITER $$
DROP FUNCTION IF EXISTS `ut_time_earth` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `ut_time_earth`(Y int,M int,D
int,HR int,MN int,SEC int) RETURNS double
    DETERMINISTIC
begin
declare T real;
set T = ROUND(367*Y - (7*(Y + FLOOR(((M+9)/12)))/4) + FLOOR((275*M)/9) +
D -730530) + (HR/24 - 0.083) + MN/1440 + SEC/86400;
return T;
end $$
DELIMITER ;
```

Function " Ut_time_epsilon()":

This function calculates the UT time epsilon from the date and UT time a picture

was taken.

```
DELIMITER $$
DROP FUNCTION IF EXISTS `ut_time_epsilon` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `ut_time_epsilon`(Y int,M
int,D int,HR int,MN int,SEC int) RETURNS double
    DETERMINISTIC
begin
declare T real;
set T = ROUND(367*Y - (7*(Y + FLOOR(((M+9)/12)))/4) + FLOOR((275*M)/9) +
D -694007) +0.5 + HR/24 + MN/1440 + SEC/86400;
return T;
end $$
DELIMITER ;
```

Function "Ut_time()":

This function calculates the decimal UT time from the date and UT time a picture

was taken.

```
DELIMITER $$

DROP FUNCTION IF EXISTS `ut_time` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `ut_time`(Y int,M int,D int,HR
int,MN int,SEC int) RETURNS double
    DETERMINISTIC
begin
declare T real;
set T = ROUND(367*Y - (7*(Y + FLOOR(((M+9)/12)))/4) + FLOOR((275*M)/9) +
D -730530) + HR/24 + MN/1440 + SEC/86400;
return T;
end $$
DELIMITER ;
```

Function " ec_anomaly()":

This function calculates the eccentricity anomaly from the mean anomaly, the eccentricity and a tolerance value of 0.0000001 of each asteroid at the date and time of each picture.

```
DELIMITER $$
DROP FUNCTION IF EXISTS `f_geo_ecliptic_long` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `f_geo_ecliptic_long`(le_in
real,rp_in real,re_in real,lp_in real) RETURNS double
    DETERMINISTIC
begin
DECLARE v_pi real;
DECLARE v_lambda real;
  SET v_pi=PI();
  IF (rp_in < re_in ) THEN
    SET v_lambda=180 + le_in + 180/v_pi*ATAN((rp_in*SIN(RADIANS(le_in -
lp_in)))/(re_in - rp_in*COS(RADIANS(le_in-lp_in))));
  ELSE
    SET v_lambda=lp_in+180/v_pi*ATAN((re_in*SIN(RADIANS(lp_in-
le_in)))/(rp_in-re_in*COS(RADIANS(lp_in-le_in)))) + 360;
  END IF;
return v_lambda;
end $$
DELIMITER ;
```

Function `f_heliocentric_dist ()":

Function "f_heliocentric_dist ":

This function calculates the heliocentric distance of each asteroid at the date and time of the picture from the mean distance, the eccentricity and the true anomaly of this asteroid.

```
DELIMITER $$

DROP FUNCTION IF EXISTS `f_heliocentric_dist` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `f_heliocentric_dist`(a_in
real,e_in real,v_in real) RETURNS double
    DETERMINISTIC
begin
declare v_pi real;
declare v_Num real;
declare v_Den real;
declare v_r real;
SET v_pi=PI();

  SET v_Num=a_in*(1-e_in*e_in);
  SET v_Den=1+e_in*cos(RADIANS(v_in));
  SET v_r=v_Num/v_Den;

return v_r;
end $$

DELIMITER ;
```

Function `f_heliocentric_lat ()":

This function calculates the heliocentric latitude of each asteroid at the date and time of the picture from the eliocentric longitude, the long ascending node and the orbit inclination of the asteroid.

```
DELIMITER $$

DROP FUNCTION IF EXISTS `f_heliocentric_lat` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `f_heliocentric_lat`(l_in
real,o_in real,i_in real) RETURNS double
    DETERMINISTIC
begin
declare v_phi real;
declare v_pi real;

  SET v_pi=PI();

  SET v_phi=(180/v_pi)*asin(sin(RADIANS(l_in-o_in))*sin(RADIANS(i_in)));

return v_phi;
end $$

DELIMITER ;
```

Function ` f_true_anomaly ()":

This function calculates the true anomaly of each asteroid at the date and time of the picture from the eccentricity and the anomaly of the eccentricity for this asteroid.

```
DELIMITER $$

DROP FUNCTION IF EXISTS `f_true_anomaly` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `f_true_anomaly`(E_in real,e
real) RETURNS double
    DETERMINISTIC
begin
declare v_pi real;
declare v_arg1 real;
declare v_arg2 real;
declare V real;
SET v_pi=PI();

  SET v_arg1=sqrt(1-e)*cos(RADIANS(E_in/2));
  SET v_arg2=sqrt(1+e)*sin(RADIANS(E_in/2));
  SET V=180/v_pi*2*ATAN2(v_arg2,v_arg1);

return V;
end $$

DELIMITER ;
```

Function ` `f_geo_ecliptic_lat ()":

This function calculates the geocentric ecliptic latitude of each asteroid at the date and time of the picture from the mean longitude of Earth, the heliocentric distance projection, the heliocentric distance Earth the heliocentric longitude projection, the heliocentric latitude and the geocentric ecliptic longitude for this asteroid.

```
DELIMITER $$

DROP FUNCTION IF EXISTS `f_geo_ecliptic_lat` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `f_geo_ecliptic_lat`(le_in
real,rp_in real,re_in real,lp_in real,phe_in real,lambda_in real) RETURNS double
    DETERMINISTIC
begin
DECLARE v_pi real;
DECLARE v_beta real;

  SET v_pi=PI();
  SET
v_beta=180/v_pi*ATAN((rp_in*TAN(RADIANS(phe_in))*SIN(RADIANS(lambda_i
n - lp_in)))/(re_in*SIN(RADIANS(lp_in-le_in))));
  return v_beta;

end $$

DELIMITER ;
```

Function "f_geo_ecliptic_long()":

This function calculates the geocentric ecliptic longitude of each asteroid at the date and time of the picture from the mean longitude of Earth, the heliocentric distance projection, the heliocentric distance Earth the heliocentric longitude projection for this asteroid.

```
DELIMITER $$

DROP FUNCTION IF EXISTS `f_geo_ecliptic_long` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `f_geo_ecliptic_long`(le_in
real,rp_in real,re_in real,lp_in real) RETURNS double
    DETERMINISTIC
begin
DECLARE v_pi real;
DECLARE v_lambda real;

  SET v_pi=PI();
  IF (rp_in < re_in ) THEN
    SET v_lambda=180 + le_in + 180/v_pi*ATAN((rp_in*SIN(RADIANS(le_in -
lp_in)))/(re_in - rp_in*COS(RADIANS(le_in-lp_in))));
  ELSE
    SET v_lambda=lp_in+180/v_pi*ATAN((re_in*SIN(RADIANS(lp_in-
le_in)))/(rp_in-re_in*COS(RADIANS(lp_in-le_in)))) + 360;
  END IF;

return v_lambda;

end $$

DELIMITER ;
```

Procedure "pic_RA_CALC":

This procedure re-calculates the RA decimal value of the center of each picture in the "tbl_picture" table.

```
DELIMITER $$

DROP PROCEDURE IF EXISTS `pic_RA_CALC` $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `pic_RA_CALC`()
BEGIN
DECLARE v_p1,b INT;
DECLARE v_RA2000 varchar(50);
DECLARE v_RA_CALC_HR,v_RA_CALC_MIN INT;
DECLARE v_RA_CALC_SEC FLOAT;
DECLARE v_RA_CALC REAL(15,12);
DECLARE cur_1 CURSOR FOR SELECT p1,RA2000 from tbl_pictures;
DECLARE CONTINUE HANDLER FOR NOT FOUND
SET b=1;
OPEN cur_1;
  REPEAT
  SET v_p1=NULL;
  FETCH cur_1 INTO v_p1,v_RA2000;
  SET v_RA_CALC_HR = substr(v_RA2000,-12,2);
  SET v_RA_CALC_MIN = substr(v_RA2000,-9,2);
  SET v_RA_CALC_SEC = substr(v_RA2000,-6,6);
  SET v_RA_CALC=ROUND((v_RA_CALC_HR + v_RA_CALC_MIN/60 +
v_RA_CALC_SEC/3600)*15,4);
  UPDATE tbl_pictures SET RA_CALC = v_RA_CALC where p1=v_p1;
UNTIL b=1
  END REPEAT;
CLOSE cur_1;
END $$

DELIMITER ;
```

Procedure "pic_DEC_CALC":

This procedure re-calculates the DEC decimal value of the center of each picture in the "tbl_picture" table.

```
DELIMITER $$

DROP PROCEDURE IF EXISTS `pic_DEC_CALC` $$
CREATE DEFINER=`root`@`localhost` PROCEDURE `pic_DEC_CALC`()
BEGIN
DECLARE v_p1,b INT;
DECLARE v_DEC2000 varchar(50);
DECLARE v_SIGN varchar(1);
DECLARE v_DEC_CALC_HR,v_DEC_CALC_MIN INT;
DECLARE v_DEC_CALC_SEC FLOAT;
DECLARE v_DEC_CALC REAL(15,12);
DECLARE cur_1 CURSOR FOR SELECT p1,DEC2000 from tbl_pictures;
DECLARE CONTINUE HANDLER FOR NOT FOUND
SET b=1;
OPEN cur_1;
  REPEAT
  SET v_p1=NULL;
  FETCH cur_1 INTO v_p1,v_DEC2000;
  SET v_SIGN= substr(v_DEC2000,-12,1);
  SET v_DEC_CALC_HR = substr(v_DEC2000,-11,2);
  SET v_DEC_CALC_MIN = substr(v_DEC2000,-8,2);
  SET v_DEC_CALC_SEC = substr(v_DEC2000,-5,6);
  SET v_DEC_CALC=ROUND((v_DEC_CALC_HR + v_DEC_CALC_MIN/60 +
v_DEC_CALC_SEC/3600),4);
  IF v_SIGN='-' THEN SET v_DEC_CALC=-v_DEC_CALC;
  END IF;
  UPDATE tbl_pictures SET DEC_CALC = v_DEC_CALC where p1=v_p1;
UNTIL b=1
  END REPEAT;
CLOSE cur_1;
END $$

DELIMITER ;
```

Function " f_magnitude()":

This function calculates the apparent magnitude of each asteroid at the date and time of the picture from the mean distance Earth, the heliocentric distance, the mean longitude Earth, the heliocentric longitude and the absolute magnitude of this asteroid.

```
DELIMITER $$

DROP FUNCTION IF EXISTS `f_magnitude` $$
CREATE DEFINER=`root`@`localhost` FUNCTION `f_magnitude`(ae_in
real,ra_in real,le_in real,l_in real,H_in float) RETURNS double
    DETERMINISTIC
begin
declare v_pi real;
declare v_delta real;
declare v_phase_effect real;
declare v_magnitude real;
SET v_pi=PI();

  SET v_delta=SQRT(ae_in*ae_in + ra_in*ra_in -
2*ra_in*ae_in*COS(RADIANS(le_in-l_in)));
  SET v_phase_effect=0.04*(180/v_pi)*ABS(ASIN(
(ae_in/v_delta)*SIN(RADIANS(le_in-l_in)))       );
  SET v_magnitude=H_in + 5*log10(ra_in*v_delta) + v_phase_effect;
return v_magnitude;
end $$

DELIMITER ;
```

APPENDIX D: PHP PROGRAMS

This Php program called "aster_coordinate.php" is published on line at:

http://ucfphysics.whsites.net/ucf.php

Under the first option, the program sends email requests to the Horizons system for each asteroid found in the "tbl_matches" table.

Under another option, the email replies data from Horizons is re-imported into the tbl_matches table and the table is updated with this Horizons data.

```
<?
include "include/session.php";
include "include/z_db.php";
?>

<html>
<HEAD>
  <?
   $hour_id=$HTTP_POST_VARS['hour_id'];
   $min_id =$HTTP_POST_VARS['min_id'];
   $sec_id =$HTTP_POST_VARS['sec_id'];
   $asteroidbyid =$HTTP_POST_VARS['asteroidbyid'];
   $asteroidbyname =$HTTP_POST_VARS['asteroidbyname'];
   $asteroidid =$HTTP_POST_VARS['asteroidid'];
   $asteroidname =$HTTP_POST_VARS['asteroidname'];
   $excentricity =$HTTP_POST_VARS['excentricity'];
   $mean_distance =$HTTP_POST_VARS['mean_distance'];
   $arg_perihelion =$HTTP_POST_VARS['arg_perihelion'];
   $long_asc_node =$HTTP_POST_VARS['long_asc_node'];

   $inclin_of_orbit=$HTTP_POST_VARS['inclin_of_orbit'];
   $inclin_orbit=$HTTP_POST_VARS['inclin_orbit'];


   $epoch_osculation =$HTTP_POST_VARS['epoch_osculation'];
   $mean_anomaly1 =$HTTP_POST_VARS['mean_anomaly1'];
   $true_anomaly =$HTTP_POST_VARS['true_anomaly'];
   $heliocentric_dist=$HTTP_POST_VARS['heliocentric_dist'];
   $heliocentric_lat=$HTTP_POST_VARS['heliocentric_lat'];
   $x_cart_ecliptic_coord=$HTTP_POST_VARS['x_cart_ecliptic_coord'];
```

```
    $y_cart_ecliptic_coord=$HTTP_POST_VARS['y_cart_ecliptic_coord'];
    $polar_ecliptic_angle=$HTTP_POST_VARS['polar_ecliptic_angle'];
    $heliocentric_long_proj=$HTTP_POST_VARS['heliocentric_long_proj'];
    $heliocentric_dist_proj=$HTTP_POST_VARS['heliocentric_dist_proj'];
    $selectasteroid =$HTTP_POST_VARS['selectasteroid'];
    $date_in=$HTTP_POST_VARS['date_in'];
    $calc_procedure_flag=$HTTP_POST_VARS['calc_procedure_flag'];
    $calc_picture_flag=$HTTP_POST_VARS['calc_picture_flag'];
    $calc_email_flag=$HTTP_POST_VARS['calc_email_flag'];
    $match_email_flag=$HTTP_POST_VARS['match_email_flag'];
    $gen_matches_flag=$HTTP_POST_VARS['gen_matches_flag'];

    $update_matches_flag=$HTTP_POST_VARS['update_matches_flag'];


    $right_ascention_calc=$HTTP_POST_VARS['right_ascention_calc'];
    $declination_calc=$HTTP_POST_VARS['declination_calc'];
    $julian_days_calc=$HTTP_POST_VARS['julian_days_calc'];
    $julian_century_time_calc=$HTTP_POST_VARS['julian_century_time_calc'];

$epsilon_century_time_calc=$HTTP_POST_VARS['epsilon_century_time_calc'];

   ?>

   <?
    function
update_match($object_name,$RA,$DEC,$RA_VAL,$DEC_VAL,$MAG,$sline,$pi
c_time)
    {
     $update_sql1 = "update tbl_matches set
HORIZONS_RA='$RA',HORIZONS_DEC='$DEC' where
AST_name='$object_name' and PIC_UT_STR='$pic_time';";
     $update_sql2 = "update tbl_matches set HORIZONS_LINE='$sline' where
AST_name='$object_name' and PIC_UT_STR='$pic_time';";
     $update_sql3 = "update tbl_matches set
HORIZONS_RA_VAL=$RA_VAL,HORIZONS_DEC_VAL=$DEC_VAL where
AST_name='$object_name' and PIC_UT_STR='$pic_time';";
     $update_sql4 = "update tbl_matches set HORIZONS_MAG=$MAG where
AST_name='$object_name' and PIC_UT_STR='$pic_time';";
     $update_sql5 = "update tbl_matches set
HORIZ_DELTA_RA=abs(AST_RA_CALC-$RA_VAL) where
AST_name='$object_name' and PIC_UT_STR='$pic_time';";
```

```
    $update_sql6 = "update tbl_matches set
HORIZ_DELTA_DEC=abs(AST_DEC_CALC-$DEC_VAL) where
AST_name='$object_name' and PIC_UT_STR='$pic_time';";
    $update_sql7 = "update tbl_matches set
PIC_DELTA_RA=abs(PIC_RA_CALC-$RA_VAL) where
AST_name='$object_name' and PIC_UT_STR='$pic_time';";
    $update_sql8 = "update tbl_matches set
PIC_DELTA_DEC=abs(PIC_DEC_CALC-$DEC_VAL) where
AST_name='$object_name' and PIC_UT_STR='$pic_time';";

    $result1 = mysql_query($update_sql1);
    $result2 = mysql_query($update_sql2);
    $result3 = mysql_query($update_sql3);
    $result4 = mysql_query($update_sql4);
    $result5 = mysql_query($update_sql5);
    $result6 = mysql_query($update_sql6);
    $result7 = mysql_query($update_sql7);
    $result8 = mysql_query($update_sql8);
    if (!$result1) {echo $update_sql1;}
    if (!$result2) {echo $update_sql2;}
    if (!$result3) {echo $update_sql3;}
    if (!$result4) {echo $update_sql4;}
    if (!$result5) {echo $update_sql5;}
    if (!$result6) {echo $update_sql6;}
    if (!$result7) {echo $update_sql7;}
    if (!$result8) {echo $update_sql8;}
   }

function convert_RA($RA)
{
$HOUR=SUBSTR($RA,0,2);
$MIN=SUBSTR($RA,3,2);
$SEC=SUBSTR($RA,6,7);
//echo "Hours=".$HOUR;print "<br>";
//echo "min=".$MIN;print "<br>";
//echo "sec=".$SEC;print "<br>";
$RA_VAL=ROUND(($HOUR+$MIN/60+$SEC/3600)*15,4);
//echo "RA_VAL=".$RA_VAL;print "<br>";
return $RA_VAL;
}

function convert_DEC($DEC)
{
$SIGN=SUBSTR($DEC,0,1);
```

```
$HOUR=SUBSTR($DEC,1,2);
$MIN=SUBSTR($DEC,4,2);
$SEC=SUBSTR($DEC,7,7);
//echo "Sign=".$SIGN;print "<br>";
//echo "Hours=".$HOUR;print "<br>";
//echo "min=".$MIN;print "<br>";
//echo "sec=".$SEC;print "<br>";
if ($SIGN=='-')
{$DEC_VAL=-ROUND($HOUR+$MIN/60+$SEC/3600,5);}
else
{$DEC_VAL=ROUND($HOUR+$MIN/60+$SEC/3600,5);}
//echo "DEC_VAL=".$DEC_VAL;print "<br>";
return $DEC_VAL;
}

?>

  <?
   function p_right_ascention($RA)
   {
    //$pi=pi();
    //$RA_HR=FLOOR($lambda/15);
    //$RA_MIN=FLOOR(($lambda/15-$RA_HR)*60);
    //$RA_SEC=ROUND((($lambda/15-$RA_HR)*60) - $RA_MIN)*60,1);
    //$RA=$RA_HR.":".$RA_MIN.":".$RA_SEC;
    //return $RA;
    $RA_CALC=substr($RA,0,2);
    return $RA_CALC;
   }

  ?>

  <?
   if ($calc_picture_flag=='Y')
   {
      $p=1;
      for ($p=1;$p<=8;$p++)
      {
       $select_pict = "SELECT p1,FRAME_ID,RA2000,DEC2000 from
tbl_pictures where p1=".$p."";
       $rs = mysql_query($select_pict);
       while($row = mysql_fetch_array($rs))
       {
        $pictureid=$row['p1'];
```

```
        $frame_id=$row['FRAME_ID'];
        $ra2000=$row['RA2000'];
        $dec2000=$row['DEC2000'];
       }
       $RA_CALC=p_right_ascention($ra2000);
       $qry="update tbl_pictures set RA_CALC=$RA_CALC,DEC_CALC=$p
where p1=".$p."";
       echo $qry;
       $rt=mysql_query($qry);
      }
      $calc_picture_flag='N';
    }
  ?>

  <?
    function
email_to_horizon($email_from,$body_name,$START_TIME,$STOP_TIME)
    {
        $email_message ="!$\$SOF";
        $email_message .="\r\n";
      $email_message .="COMMAND ='".$body_name."'";
        $email_message .="\r\n";
      $email_message .="OBJ_DATA ='YES' ! No summary of target body data ";
        $email_message .="\r\n";
      $email_message .="MAKE_EPHEM ='YES' ! Make an ephemeris ";
        $email_message .="\r\n";
      $email_message .="START_TIME ='".$START_TIME."' ! Start of table (UTC
default) ";
        $email_message .="\r\n";
      $email_message .="STOP_TIME ='".$STOP_TIME."' ! End of table (UTC
default) ";
        $email_message .="\r\n";
      $email_message .="STEP_SIZE = '2 day' ! Table step-size ";
        $email_message .="\r\n";
      //$email_message .="TABLE_TYPE = 'OBS' ! Specify VECTOR ephemeris
table type ";
      $email_message .="CENTER= '568' ! Center-site name: Mauna Kea ";
        $email_message .="\r\n";
      $email_message .="REF_PLANE = 'FRAME' ! J2000 equatorial plane ";
        $email_message .="\r\n";
      $email_message .="QUANTITIES = '1,3,9' ! Compute these quantities (See
doc) ";
        $email_message .="\r\n";
```

```
      $email_message .="ANG_FORMAT = 'HMS' ! Format of RA/DEC (HMS or
DEG) ";
        $email_message .="\r\n";
      $email_message .="EXTRA_PREC = 'YES' ! Additional output digits of
RA/DEC ";
        $email_message .="\r\n";
      $email_message .="CSV_FORMAT = 'NO' ! Comma-separated output
(YES/NO) ";
        $email_message .="\r\n";
        $email_message .= "!$\$EOF";

      $email_to_ME='horizons@ssd.jpl.nasa.gov';
      $email_subject='JOB';
      mail( $email_to_ME, "JOB",$email_message, "From: $email_from" );
    }
  ?>

  <?
    if ($calc_procedure_flag=='Y')
    {

     $select_astid_min_cnt = "select min(f1) as cnt_min from tbl_astorb where
DEC_calc is null";
     $rs_min_cnt = mysql_query($select_astid_min_cnt);
     while($row = mysql_fetch_array($rs_min_cnt))
     {
       $cnt_min=$row['cnt_min'];
     }

     $select_astid_max_cnt = "select max(f1) as cnt_max from tbl_astorb";
     $rs_max_cnt = mysql_query($select_astid_max_cnt);
     while($row = mysql_fetch_array($rs_max_cnt))
     {
       $cnt_max=$row['cnt_max'];
     }


     echo " cnt_min=".$cnt_min;
     echo " cnt_max=".$cnt_max;

     for ($j=$cnt_min;$j<=$cnt_max;$j++)
     {
```

```
	$select_astid = "SELECT f1,f2,f12,f13,f14,f15,f16,f17,f18 from tbl_astorb
where f1=".$j."";
	$rs = mysql_query($select_astid);

	$k=0;
	while($row = mysql_fetch_array($rs))
	{
	 $k++;
	 $asteroidid=$row['f1'];
	 $asteroidname=$row['f2'];
	 $epoch_osculation=$row['f12'];
	 $mean_anomaly1=$row['f13'];
	 $arg_perihelion=$row['f14'];
	 $long_asc_node=$row['f15'];
	 $inclin_of_orbit=$row['f16'];
	 $inclin_orbit=$row['f16'];
	 $excentricity=$row['f17'];
	 $mean_distance=$row['f18'];
	}
	list($year_eo,$month_eo,$day_eo) = explode("-", $epoch_osculation);
	$epoch_osculation=$month_eo."/".$day_eo."/".$year_eo;
	$ut2=ut_time($year_eo,$month_eo,$day_eo,0,0,0);
	$R= pow($mean_distance, 3)*1.3341260180*pow(10, 5);
	$P=pow($R, 0.5);
	$alpha1=360/$P;

	$ut=$julian_days_calc;
	//echo " ut=".$ut;
	$DT=$ut-$ut2;
	$M2 = $mean_anomaly1 + $alpha1*($ut-$ut2);
	$mean_anomaly2=rev($M2);

	$A=$arg_perihelion + $long_asc_node;
	$long_perihelion=rev($A);
	$E_tolerance=0.0000001;

$excentric_anomaly=ec_anomaly($mean_anomaly2,$excentricity,$E_tolerance);

	$E=$excentric_anomaly;
	$e=$excentricity;
	$v=f_true_anomaly($E,$e);
	$true_anomaly=rev($v);

	$l=$true_anomaly+$long_perihelion;
```

```
$heliocentric_long=rev($l);

$a=$mean_distance;
$e=$excentricity;
$v=$true_anomaly;
$heliocentric_dist=f_heliocentric_dist($a,$e,$v);

$l=$heliocentric_long;
$O=$long_asc_node;
$i=$inclin_orbit;
$phi=f_heliocentric_lat($l,$O,$i);
$heliocentric_lat=rev($phi);

$l=$heliocentric_long;
$O=$long_asc_node;
$x_cart_ecliptic_coord=cos(deg2rad($l-$O));

$i=$inclin_orbit;
$l=$heliocentric_long;
$O=$long_asc_node;
$y_cart_ecliptic_coord=sin(deg2rad($l-$O))*cos(deg2rad($i));


$phi_ecliptic=180/PI()*ATAN2($y_cart_ecliptic_coord,$x_cart_ecliptic_coord);
$polar_ecliptic_angle=rev($phi_ecliptic);

$O=$long_asc_node;
$phi=$polar_ecliptic_angle;
$lp=$phi+$O;
$heliocentric_long_proj=rev($lp);

$r=$heliocentric_dist;
$phi=$heliocentric_lat;
$heliocentric_dist_proj=$r*COS(deg2rad($phi));

$ao=1.00000011;
$mean_distance_earth_o=$ao;
$eo=0.01671022;
$excentricity_earth_o=$eo;
$io=2.77274871;
$inclin_orbit_earth_o=$io;
$Wo=102.94719;
$long_perihelion_earth_o=$Wo;
$Oo=348.73936;
```

```
$long_asc_node_earth_o=$Oo;
$lo=100.46435;
$mean_long_earth_o=$lo;
$ar=-5/100000000;
$mean_distance_earth_r=$ar;
$er=-0.00003804;
$excentricity_earth_r=$er;
$ir=-46.94;
$inclin_orbit_earth_r=$ir;
$Wr=1198.28;
$long_perihelion_earth_r=$Wr;
$Or=-18228.25;
$long_asc_node_earth_r=$Or;
$lr=1293740.63;
$mean_long_earth_r=$lr;
$Nr=99;
$mean_long_earth_factor_r=$Nr;

$t=$julian_century_time_calc;

$at=$ao+$ar*$t;
$mean_distance_earth_t=$at;
$et=$eo + $er*$t;
$excentricity_earth_t=$et;
$Wt=$Wo + ($Wr / 3600)*$t;
$long_perihelion_earth_t=$Wt;

$Ot=$Oo + ($Or / 3600)*$t;
$long_asc_node_earth_t=$Ot;

$lt=$lo + ($lr / 3600 + 360*$Nr)*$t;
$mean_long_earth_t=rev($lt);

$lt=$mean_long_earth_t;
$Mt=$lt-$Wt;
$mean_anomaly_earth_t=$Mt;

$E_tolerance=0.0000001;

$exc_anomaly_earth_t=ec_anomaly($mean_anomaly_earth_t,$excentricity_eart
h_t,$E_tolerance);

$Et=$exc_anomaly_earth_t;
$et=$excentricity_earth_t;
```

```
	$vt=f_true_anomaly($Et,$et);
	$true_anomaly_earth_t=$vt;
	$lp=$heliocentric_long_proj;

	$heliocentric_dist_earth_t=($at*(1-$et*$et))/(1+$et*COS(deg2rad($vt)));

	$le=$mean_long_earth_t;
	$rp=$heliocentric_dist_proj;
	$re=$heliocentric_dist_earth_t;
	$lp=$heliocentric_long_proj;

	$lambda=f_geo_ecliptic_long($le,$rp,$re,$lp);
	$geo_ecliptic_long=rev($lambda);

	$le=$mean_long_earth_t;
	$rp=$heliocentric_dist_proj;
	$re=$heliocentric_dist_earth_t;
	$lp=$heliocentric_long_proj;
	$phi=$heliocentric_lat;

	$lambda=$geo_ecliptic_long;
	$beta=f_geo_ecliptic_lat($le,$rp,$re,$lp,$phi,$lambda);
	$geo_ecliptic_lat=rev($beta);

	$T_epsilon=$epsilon_century_time_calc;
	//echo "T_epsilon=".$T_epsilon;
	$ecliptic_mean_obliquity=23.452294 - (  46.845*$T_epsilon +
0.0059*POW($T_epsilon,2) - 0.00181*POW($T_epsilon,3)  ) / 3600;

	$pi=pi();
	$epsilon=$ecliptic_mean_obliquity;
	$beta=$geo_ecliptic_lat;
	$lambda=$geo_ecliptic_long;

$SIN_DELTA=SIN(deg2rad($beta))*COS(deg2rad($epsilon))+COS(deg2rad($be
ta))*SIN(deg2rad($epsilon))*SIN(deg2rad($lambda));
	$declination=180/$pi*ASIN($SIN_DELTA);

	$pi=pi();
	$epsilon=$ecliptic_mean_obliquity;
	$beta=$geo_ecliptic_lat;
	$lambda=$geo_ecliptic_long;
	$alpha_y=SIN(deg2rad($lambda))*COS(deg2rad($epsilon)) -
TAN(deg2rad($beta))*SIN(deg2rad($epsilon));
```

```
        //=COS(RADIANS(lambda))
        $alpha_x=COS(deg2rad($lambda));
        //$right_ascention=f_right_ascension($lambda);
        $RA=180/$pi*ATAN2($alpha_y,$alpha_x);
        $right_ascention=rev($RA);

        $right_ascention_calc=$right_ascention;
        $declination_calc=$declination;
        $qry="update tbl_astorb set
DEC_calc=$declination_calc,RA_calc=$right_ascention_calc where f1=".$j."";
        //echo $qry;
        $rt=mysql_query($qry);
      }

      $calc_procedure_flag='N';
      if (mysql_error()<>'')
      {
        $error_msg=mysql_error();
        echo $error_msg;
      }
     }
   ?>

   <?
     if ($selectasteroid=='byid')
     {
       $select_astid = "SELECT f1,f2,f12,f13,f14,f15,f16,f17,f18 from tbl_astorb
where f1=".$asteroidbyid."";
       //echo $select_astid;
       $rs = mysql_query($select_astid);
       $i=0;
       while($row = mysql_fetch_array($rs))
       {
        $i++;
        $asteroidid=$row['f1'];
        $asteroidname=$row['f2'];
        $epoch_osculation=$row['f12'];
        $mean_anomaly1=$row['f13'];
        $arg_perihelion=$row['f14'];
        $long_asc_node=$row['f15'];
        $inclin_of_orbit=$row['f16'];
        $inclin_orbit=$row['f16'];
        $excentricity=$row['f17'];
        $mean_distance=$row['f18'];
```

```
      }
      list($year_eo,$month_eo,$day_eo) = explode("-", $epoch_osculation);
      $epoch_osculation=$month_eo."/".$day_eo."/".$year_eo;
     }

    if ($selectasteroid=='byname')
    {
      $select_astid = "SELECT f1,f2,f12,f13,f14,f15,f16,f17,f18 from tbl_astorb
where f2='".$asteroidbyname."'";
      //echo $select_astid;
      $rs = mysql_query($select_astid);
      $i=0;
      while($row = mysql_fetch_array($rs))
      {
       $i++;
       $asteroidid=$row['f1'];
       $asteroidname=$row['f2'];
       $epoch_osculation=$row['f12'];
       $mean_anomaly1=$row['f13'];
       $arg_perihelion=$row['f14'];
       $long_asc_node=$row['f15'];
       $inclin_of_orbit=$row['f16'];
       $inclin_orbit=$row['f16'];
       $excentricity=$row['f17'];
       $mean_distance=$row['f18'];
      }
      list($year_eo,$month_eo,$day_eo) = explode("-", $epoch_osculation);
      $epoch_osculation=$month_eo."/".$day_eo."/".$year_eo;
     }

     echo "<tr><td width=20 height=20></td></tr>";
  ?>


  <script type="text/javascript" src="scripts/calendar_date_picker.js"></script>
  <script type="text/javascript">
        var cdp1 = new CalendarDatePicker();
      var props =
      {
          debug :                 true,
          minimumFutureDate :     5,
          formatDate :            '%m/%d/%y'
        };
      var cdp2 = new CalendarDatePicker(props);
```

```
    props.formatDate = '%m/%d/%y';
    var cdp3 = new CalendarDatePicker(props);
    cdp3.endYear = cdp3.startYear + 1;
  </script>

  <script>
      function setWidth()
      {
    document.getElementById("date_in").style.height="20px";
      document.getElementById("date_in").style.width="75px";
      document.getElementById("asteroidbyid").style.height="20px";
      document.getElementById("asteroidbyid").style.width="75px";

      document.getElementById("button4").style.height="30px";
      document.getElementById("button4").style.width="260px";
      document.getElementById("button5").style.height="30px";
      document.getElementById("button5").style.width="260px";
      document.getElementById("button6").style.height="30px";
      document.getElementById("button6").style.width="260px";
      document.getElementById("button7").style.height="30px";
      document.getElementById("button7").style.width="260px";
      }

      function select_ast_id()
      {
    document.getElementById("selectasteroid").value='byid';
    document.getElementById("asteroidbyname").value='';
    }

      function select_ast_name()
      {
    document.getElementById("selectasteroid").value='byname';
    document.getElementById("asteroidbyid").value='';
    }

  </script>
</HEAD>

<?



function ut_time($Y,$M,$D,$HR,$MIN,$SEC)
{
```

```
 //$T=367*$Y - (7*($Y + (($M+9)/12)))/4 + (275*$M)/9 + $D -730530 + $HR/24 +
$MIN/1440 + $SEC/86400;
 $T=ROUND(367*$Y - (7*($Y + FLOOR((($M+9)/12)))/4) + FLOOR((275*$M)/9)
+ $D -730530) + $HR/24 + $MIN/1440 + $SEC/86400;
 return $T;
}

function ut_time_earth($Y,$M,$D,$HR,$MIN,$SEC)
{
 $T=ROUND(367*$Y - (7*($Y + FLOOR((($M+9)/12)))/4) + FLOOR((275*$M)/9)
+ $D -730530)  -0.5 + $HR/24 + $MIN/1440 + $SEC/86400;
 //echo "T=".$T;
 return $T;
}

function ut_time_epsilon($Y,$M,$D,$HR,$MIN,$SEC)
{
 // =(ROUND(367*year - ( 7* (year+ FLOOR((  (month+9)/12  ),1)     )) /4 +
FLOOR((275*month)/9,1) + day - 694007,0) + 0.5 + hour/24 + minute/1440 +
seconde/86400)/E21
 $T=ROUND(367*$Y - (7*($Y + FLOOR((($M+9)/12)))/4) + FLOOR((275*$M)/9)
+ $D -694007)  +0.5 + $HR/24 + $MIN/1440 + $SEC/86400;
 //echo "T=".$T;
 return $T;
}


function ec_anomaly($M,$e,$E_tolerance)
{
  $pi=pi();
  $E=1;
  $error=100;
  $i=0;
  while ($error>=$E_tolerance and $i<=100)
  {
     //Excel
     //=C6  - (      (M   -   C6    +      (180/PI())*e*SIN(RADIANS(C6))) /
(e*COS(RADIANS(C6))-1)      )

     //php
     $E=$E-(       ($M-$E+   (180/$pi)*$e*SIN(deg2rad($E))
)/($e*COS(deg2rad($E))-1)   );

     //Excel
```

```
    //  =M+e*SIN(RADIANS(C7))*(180/PI())

    //php
    $E_check= $M+$e*SIN(deg2rad($E))*(180/$pi);

    $error=abs($E-$E_check);
    //echo " i=".$i;
    //echo " error=".$error;
    //echo " E".$i."=".$E;
    $i++;
  }
 return $E;
}

function f_true_anomaly($E,$e)
{
  $pi=pi();
  //echo " E=".$E;
  //echo " e=".$e;
  //v=2*arg(sqr(1-e)*cos(E/2),sqr(1+e)*sin(E/2))
  //Excel =180/PI()*2*ATAN2(SQRT(1-
e)*COS(RADIANS(C27/2)),SQRT(1+e)*SIN(RADIANS(C27/2)))
  $arg1=sqrt(1-$e)*cos(deg2rad($E/2));
  $arg2=sqrt(1+$e)*sin(deg2rad($E/2));
  $v=180/PI()*2*ATAN2($arg2,$arg1);
  return $v;
}

function f_heliocentric_dist($a,$e,$v)
{
  $pi=pi();
  $Num=$a*(1-$e*$e);
  $Den=1+$e*cos(deg2rad($v));
  $r=$Num/$Den;
  return $r;
}

function f_heliocentric_lat($l,$O,$i)
{
  $pi=pi();
  $phi=(180/$pi)*asin(sin(deg2rad($l-$O))*sin(deg2rad($i)));
  //$A=sin(deg2rad($l-$O));
  //$B=sin(deg2rad($i));
  //$phi=asin($A*$B);
```

```
  return $phi;
}

function f_geo_ecliptic_long($le,$rp,$re,$lp)
{
  $pi=pi();
  if ($rp < $re )
  {
    //=180 + le + 180/PI()*ATAN((r_p*SIN(RADIANS(le - l_p)))/(re  -
r_p*COS(RADIANS(le-l_p))))
    $lambda=180 + $le + 180/$pi*ATAN(($rp*SIN(deg2rad($le - $lp)))/($re -
$rp*COS(deg2rad($le-$lp))));
  }
  else
  {
    //=l_p+180/PI()*ATAN((re*SIN(RADIANS(l_p-le)))/(r_p-re*COS(RADIANS(l_p-
le)))) + 360
    $lambda=$lp+180/$pi*ATAN(($re*SIN(deg2rad($lp-$le)))/($rp-
$re*COS(deg2rad($lp-$le)))) + 360;
  }
  return $lambda;
}

function f_geo_ecliptic_lat($le,$rp,$re,$lp,$phe,$lambda)
{
  $pi=pi();
  //=180/PI()*ATAN((r_p*TAN(RADIANS(phe))*SIN(RADIANS(lambda -
l_p)))/(re*SIN(RADIANS(l_p-le))))
  $beta=180/$pi*ATAN(($rp*TAN(deg2rad($phe))*SIN(deg2rad($lambda -
$lp)))/($re*SIN(deg2rad($lp-$le))));
  return $beta;
}

function f_right_ascention($lambda)
{
  $pi=pi();
  $RA_HR=FLOOR($lambda/15);
  $RA_MIN=FLOOR(($lambda/15-$RA_HR)*60);
  //=ROUND(((D51/15-D54)*60 - D55)*60,1)
  $RA_SEC=ROUND((($lambda/15-$RA_HR)*60) - $RA_MIN)*60,1);
  $RA=$RA_HR.":".$RA_MIN.":".$RA_SEC;
  return $RA;
}
```

```
function f_declination($delta)
{
  $pi=pi();
  if ($delta<0)
  {
   $delta_degree=-FLOOR(-$delta);
  }
  else
  {
   $delta_degree=FLOOR($delta);
  }
  $delta_minute=FLOOR((ABS($delta)-ABS($delta_degree))*60);
  $delta_second=ROUND(((ABS($delta)-ABS($delta_degree))*60 -
ABS($delta_minute))*60,1);
  $delta_format=$delta_degree.":".$delta_minute.":".$delta_second;
  return $delta_format;
}


function rev($val_in)
{
 $rv=$val_in - ROUND($val_in/360)*360;
 if ($rv<0) {$rv=$rv+360;}
 return $rv;
}


$pi=pi();
//echo $pi;

$radeg=180/$pi;
//echo $radeg;

//arcsin
$x=0.5;
$arcsin=asin($x);
//echo $arcsin;

$rev_val=rev(-3543);
//echo $rev_val;

?>

<script>
```

```
function calc_procedure()
{
 document.getElementById('calc_procedure_flag').value  ='Y';
 document.getElementById('right_ascention_calc').value
=document.getElementById('right_ascention').value;
 document.getElementById('declination_calc').value
=document.getElementById('declination').value;
 document.getElementById('julian_days_calc').value
=document.getElementById('julian_days').value;
 document.getElementById('julian_century_time_calc').value
=document.getElementById('julian_century_time').value;
 document.getElementById('epsilon_century_time_calc').value
=document.getElementById('epsilon_century_time').value;
}

function calc_picture_procedure()
{
 document.getElementById('calc_picture_flag').value  ='Y';
}

function calc_email()
{
 document.getElementById('calc_email_flag').value  ='Y';
}

function match_email()
{
 document.getElementById('match_email_flag').value  ='Y';
}


function gen_matches()
{
 document.getElementById('gen_matches_flag').value  ='Y';
}

function update_match()
{
 document.getElementById('update_matches_flag').value  ='Y';
}


</script>
```

```
<body>

<TABLE width=1080 height=180 bgcolor="lightgrey" cellspacing=1 cellpadding=1
BORDER=1>


<tr><td colspan=3 width=1040 height=20 valign=top align=center><b><font
face='Arial' size='4' color=green >Ephemeris of Small Bodies - Dr. Yan
Fernandez and Jean-Marc Denis last updaded: 04/15/2010</b></font></td></tr>

<td><TABLE width=520 height=120 cellpadding=0 cellspacing=0 BORDER=0>
      <tr><td width=200 height=20></td></tr>


      <form name='frm_para1' action='aster_coordinate.php' method='post'>
      <tr>
      <td colspan=1 width=400px>Enter an asteroid id:</td>
      <td><input type='text' name="asteroidbyid" id="asteroidbyid"></td>

      <td width=40px align=left>
        <input type='submit' name="button1" id="button1" value='submit'
onclick="select_ast_id();submit();"></td>

      </tr>

      <tr>
      <td colspan=1 width=400px>or an asteroid name:</td>
      <td><input type='text' name="asteroidbyname" id="asteroidbyname"></td>

      <td width=40px align=left>
        <input type='submit' name="button2" id="button2" value='submit'
onclick="select_ast_name();submit();"></td>

      </tr>

      <tr>
      <td colspan=1 width=400px>Asteroid Id</td>
      <td colspan=2><input type='text' name="asteroidid" id="asteroidid"></td>
      </tr>

      <tr>
      <td colspan=1 width=400px>Asteroid Name</td>
      <td colspan=2><input type='text' name="asteroidname"
id="asteroidname"></td>
```

```
        </tr>

        <tr>
        <td colspan=1 width=400px>Argument of Perihelion w (deg)</td>
        <td colspan=2><input type='text' name="arg_perihelion"
id="arg_perihelion"></td>
        </tr>

        <tr>
        <td colspan=1 width=400px>Long of Ascending Node O-Omega (deg)</td>
        <td colspan=2><input type='text' name="long_asc_node"
id="long_asc_node"></td>
        </tr>

        <tr>
        <td colspan=1 width=400px>Inclination of Orbit i for test display</td>
        <td colspan=2><input type='text' name="inclin_orbit" id="inclin_orbit"></td>
        </tr>

        <tr>
        <td colspan=1 width=400px>Excentricity e</td>
        <td colspan=2><input type='text' name="excentricity" id="excentricity"></td>
        </tr>

        <tr>
        <td colspan=1 width=400px>Semimajor Axis a (Mean distance in AU)</td>
        <td colspan=2><input type='text' name="mean_distance"
id="mean_distance"></td>
        </tr>

        <tr>
        <td colspan=1 width=400px>Epoch of Osculation</td>
        <td colspan=2><input type='text' name="epoch_osculation"
id="epoch_osculation"></td>
        </tr>

        <tr>
        <td colspan=1 width=400px>Mean Anomaly M1 (deg)</td>
        <td colspan=2><input type='text' name="mean_anomaly1"
id="mean_anomaly1"></td>
        </tr>

        <td><input type='hidden' name="selectasteroid" id="selectasteroid"></td>
```

```
<tr><td width=200 height=20></td></tr>
<form name='frm_para2' action='aster_coordinate.php' method='post'>
<tr>
<td colspan=1 width=90px>Enter a date:</td>

  <link href="scripts/calendar.css" rel="stylesheet" type="text/css">
<div name=divname>

<td width=125px align=left>

  <input type='text' name="date_in" id="date_in"><a href="#"
onclick="cdp1.showCalendar(this,'date_in');return false;"><img name
="date_inref" id="date_inref" src="images/calendar.png" border="0"></a>
</td>

</div>
  <td>
    <table id="calendarTable">
    <tbody id="calendarTableHead">
    <tr>
    <td colspan="6" align="left">
          <select id="selectMonth">
           <option value="0">January</option>
           <option value="1">February</option>
           <option value="2">March</option>
           <option value="3">April</option>
           <option value="4">May</option>
           <option value="5">June</option>
           <option value="6">July</option>
           <option value="7">August</option>
           <option value="8">September</option>
           <option value="9">October</option>
           <option value="10">November</option>
           <option value="11">December</option>
          </select>
    </td>
    <td colspan="1" align="center"><select id="selectYear"></select></td>
    <td align="right"><a href="#" id="closeCalendarLink">X</a></td>
    </tr>
    </tbody>
    <tbody id="calendarTableDays">
    <tr id="calenderDaysIndex">
```

```
<td>Su</td><td>Mo</td><td>Tu</td><td>We</td><td>Thu</td><td>Fr</td><td>
Sa</td>
            </tr>
            </tbody>
            <tbody id="calendar"></tbody>
            </table>
          </td>

        <tr><td width=250></td><td width=30 valign=center colspan=1><font
face='Arial' size='2' >HOURS.MINS.SECS.</td></tr>
        <tr>
        <td width=280px colspan=1 align=left valign=top>Enter a time:</td>


        <td colspan=1 width=150px valign=top align=left>
        <select name="hour_id" id="hour_id">
          <?php
        for ($i=0;$i<=24;$i++)
          {
          echo "<option value=".$i.">".$i;
          }
          ?>
          </select>


        <select name="min_id" id="min_id">
          <?php
        for ($i=0;$i<=60;$i++)
          {
          echo "<option value=".$i.">".$i;
          }
          ?>
          </select>

        <select name="sec_id" id="sec_id">
          <?php
        for ($i=0;$i<=60;$i++)
          {
          echo "<option value=".$i.">".$i;
          }
          ?>
          </select>
```

```
      </td>

      <td width=40px align=left>
        <input type='submit' name="button3" id="button3" value='submit'
onclick="submit();"></td>

    <tr><td width=20 height=10></td></tr>

    <tr>

    <tr width=210px colspan=3 align=right>
        <input type='submit' name="button4" id="button4" value='Email this body
to H.' onclick="calc_email();submit();"></tr>

    <tr width=210px colspan=3 align=right>
        <input type='submit' name="button5" id="button5" value='Gen. match
bodies' onclick="gen_matches();submit();"></tr>

    <tr width=210px colspan=3 align=right>
        <input type='submit' name="button6" id="button6" value='Send email
matches to H.' onclick="match_email();submit();"></tr>

    <tr width=210px colspan=3 align=right>
        <input type='submit' name="button7" id="button7" value='Update match
table from H. export file' onclick="update_match();submit();"></tr>

    <td width=20><input type='hidden' name="calc_procedure_flag"
id="calc_procedure_flag"></td>
    <td width=20><input type='hidden' name="calc_picture_flag"
id="calc_picture_flag"></td>
    <td width=20><input type='hidden' name="calc_email_flag"
id="calc_email_flag"></td>

    <td width=20><input type='hidden' name="update_matches_flag"
id="update_matches_flag"></td>

    <td width=20><input type='hidden' name="gen_matches_flag"
id="gen_matches_flag"></td>
    <td width=20><input type='hidden' name="match_email_flag"
id="match_email_flag"></td>
    <td width=20><input type='hidden' name="right_ascention_calc"
id="right_ahscention_calc"></td>
    <td width=20><input type='hidden' name="declination_calc"
id="declination_calc"></td>
```

```
    <td width=20><input type='hidden' name="julian_days_calc"
id="julian_days_calc"></td>
    <td width=20><input type='hidden' name="julian_century_time_calc"
id="julian_century_time_calc"></td>
    <td width=20><input type='hidden' name="epsilon_century_time_calc"
id="epsilon_century_time_calc"></td>
     </tr>



<?
//Time test:
//$year=1990;
//$month=04;
//$day=19;
//$date_in="$month"."/".$day."/".$year;
list($month, $day, $year) = explode("/", $date_in);
echo "<tr><td width=20 height=10></td></tr>";

echo "<tr><td valign=top colspan=1>";
//echo "ut = ";
  $ut=ut_time($year,$month,$day,$hour_id,$min_id,$sec_id);
  $julian_days=$ut;
//echo $ut;
echo "</td><td colspan=2>";
//echo "UT time 2000 Jan 0.0 TDT";
echo "</td></tr>";


echo "</td></tr>";
//echo "<tr><td width=20></td></tr>";
?>

<tr>
<td colspan=1 width=400px>Number of julian days (since J2000)</td>
<td colspan=2><input type='text' name="julian_days" id="julian_days"></td>
</tr>


<?
  $ut_earth=ut_time_earth($year,$month,$day,$hour_id,$min_id,$sec_id);
  $julian_century_time=$ut_earth/36525;
?>
```

```
<tr>
<td valign=top height=20 colspan=1 width=300px>Julian century time for earth
(J2000 - 0.5) t = </td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="julian_century_time" id="julian_century_time"></td>
</tr>


<?
  $ut_epsilon=ut_time_epsilon($year,$month,$day,$hour_id,$min_id,$sec_id);
  $epsilon_century_time=$ut_epsilon/36525;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Julian cent. time for epsilon
(J1900 + 0.5) t = </td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="epsilon_century_time" id="epsilon_century_time"></td>
</tr>

<tr><td width=20 height=20></td></tr>

<?

  //=M + (360/(POWER(POWER(a, 3)*1.334126018*POWER(10, 5),0.5)))*(ut-
ut2)+360*B24
  $ut2=ut_time($year_eo,$month_eo,$day_eo,0,0,0);
  //echo " ut2=".$ut2;
  //echo " ut=".$ut;
  $R= pow($mean_distance, 3)*1.3341260180*pow(10, 5);
  //echo " R=".$R;
  $P=pow($R, 0.5);
  //echo " P=".$P;
  $alpha1=360/$P;
  //echo " alpha1=".$alpha1;
  //echo " mean_anomaly1=".$mean_anomaly1;
  $DT=$ut-$ut2;
  //echo " DT=".$DT;
  $M2 = $mean_anomaly1 + $alpha1*($ut-$ut2);
  $mean_anomaly2=rev($M2);



?>
<tr>
```

```
<td colspan=1 width=400px>Mean Anomaly at date entered M2 (in deg)</td>
<td colspan=2><input type='text' name="mean_anomaly2"
id="mean_anomaly2"></td>
</tr>

<?
  $A=$arg_perihelion + $long_asc_node;
  $long_perihelion=rev($A);
?>
<tr>
<td colspan=1 width=400px>Longitude of Perihelion W (in deg)</td>
<td colspan=2><input type='text' name="long_perihelion"
id="long_perihelion"></td>
</tr>

<?
  $E_tolerance=0.0000001;

$excentric_anomaly=ec_anomaly($mean_anomaly2,$excentricity,$E_tolerance);
?>
<tr>
<td colspan=1 width=400px>Excentric Anomaly E (in deg)</td>
<td colspan=2><input type='text' name="excentric_anomaly"
id="excentric_anomaly"></td>
</tr>

<?
  $E=$excentric_anomaly;
  $e=$excentricity;
  $v=f_true_anomaly($E,$e);
  $true_anomaly=rev($v);
?>

<tr>
<td colspan=1 width=400px>True Anomaly v (in deg)</td>
<td colspan=2><input type='text' name="true_anomaly" id="true_anomaly"></td>
</tr>

<?
  $l=$true_anomaly+$long_perihelion;
  $heliocentric_long=rev($l);
?>
<tr>
<td colspan=1 width=400px>Heliocentric longitude l (in deg)</td>
```

```
<td colspan=2><input type='text' name="heliocentric_long"
id="heliocentric_long"></td>
</tr>

<?
 $a=$mean_distance;
 $e=$excentricity;
 $v=$true_anomaly;
 $heliocentric_dist=f_heliocentric_dist($a,$e,$v);
?>
<tr>
<td colspan=1 width=400px>Heliocentric Distance r (in AU)</td>
<td colspan=2><input type='text' name="heliocentric_dist"
id="heliocentric_dist"></td>
</tr>

<?
 $l=$heliocentric_long;
 $O=$long_asc_node;
 //$i=$inclin_of_orbit;
 $i=$inclin_orbit;
 //echo " l= ".$l;
 //echo " O= ".$O;
 //echo " i= ".$i;
 $phi=f_heliocentric_lat($l,$O,$i);
 $heliocentric_lat=rev($phi)
?>
<tr>
<td colspan=1 width=400px>Heliocentric Latitude phi (in deg)</td>
<td colspan=2><input type='text' name="heliocentric_lat"
id="heliocentric_lat"></td>
</tr>

<?
 $l=$heliocentric_long;
 $O=$long_asc_node;
 $x_cart_ecliptic_coord=cos(deg2rad($l-$O));
?>
<tr>
<td colspan=1 width=400px>Cartesian Ecliptic Coord. x</td>
<td colspan=2><input type='text' name="x_cart_ecliptic_coord"
id="x_cart_ecliptic_coord"></td>
</tr>
```

```
<?
 $i=$inclin_orbit;
 $l=$heliocentric_long;
 $O=$long_asc_node;
 $y_cart_ecliptic_coord=sin(deg2rad($l-$O))*cos(deg2rad($i));
 //$y_cart_ecliptic_coord=$l-$O;
?>
<tr>
<td colspan=1 width=400px>Cartesian Ecliptic Coord. y</td>
<td colspan=2><input type='text' name="y_cart_ecliptic_coord"
id="y_cart_ecliptic_coord"></td>
</tr>

<?
 //$phi_ecliptic=(180/PI())*ATAN($y_cart_ecliptic_coord/$x_cart_ecliptic_coord);
 $phi_ecliptic=180/PI()*ATAN2($y_cart_ecliptic_coord,$x_cart_ecliptic_coord);
 $polar_ecliptic_angle=rev($phi_ecliptic);
?>
<tr>
<td colspan=1 width=400px>Polar Ecliptic Angle arctan(y/x)</td>
<td colspan=2><input type='text' name="polar_ecliptic_angle"
id="polar_ecliptic_angle"></td>
</tr>

<?
 $O=$long_asc_node;
 $phi=$polar_ecliptic_angle;
 $lp=$phi+$O;
 $heliocentric_long_proj=rev($lp);
?>
<tr>
<td colspan=1 width=400px>Heliocentric longitude l' projected (in deg)</td>
<td colspan=2><input type='text' name="heliocentric_long_proj"
id="heliocentric_long_proj"></td>
</tr>

<?
$r=$heliocentric_dist;
$phi=$heliocentric_lat;
$heliocentric_dist_proj=$r*COS(deg2rad($phi));
?>
<tr>
<td colspan=1 width=400px>Heliocentric distance r' projected (in AU)</td>
```

```
<td colspan=2><input type='text' name="heliocentric_dist_proj"
id="heliocentric_dist_proj"></td>
</tr>

</TABLE></td>

<td><TABLE width=20 height=799 cellpadding=0 cellspacing=0 BORDER=0>
  <tr><td width=20 height=97></td></tr>
</TABLE></td>

<td><TABLE width=500 height=799 cellpadding=0 cellspacing=0 BORDER=0>
  <tr><td width=500 height=27></td></tr>

<?
$ao=1.00000011;
$mean_distance_earth_o=$ao;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Semi-major axis ao at
time t=to</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="mean_distance_earth_o" id="mean_distance_earth_o"></td>
</tr>

<?
$eo=0.01671022;
$excentricity_earth_o=$eo;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Excentricity eo at time
t=to</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="excentricity_earth_o" id="excentricity_earth_o"></td>
</tr>

<?
$io=2.77274871;
$inclin_orbit_earth_o=$io;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Inclination io at time
t=to</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="inclin_orbit_earth_o" id="inclin_orbit_earth_o"></td>
```

```
</tr>

<?
$Wo=102.94719;
$long_perihelion_earth_o=$Wo;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Longitude of perihion
Wo at time t=to</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="long_perihelion_earth_o" id="long_perihelion_earth_o"></td>
</tr>

<?
$Oo=348.73936;
$long_asc_node_earth_o=$Oo;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Long acending node Oo
at time t=to</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="long_asc_node_earth_o" id="long_asc_node_earth_o"></td>
</tr>

<?
$lo=100.46435;
$mean_long_earth_o=$lo;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Mean longitude lo at
time t=to</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="mean_long_earth_o" id="mean_long_earth_o"></td>
</tr>


  <tr><td width=500 valign=top height=20></td></tr>

<?
$ar=-5/100000000;
$mean_distance_earth_r=$ar;
?>
<tr>
```

```
<td valign=top height=20 colspan=1 width=300px>Earth Semi-major axis ar rate
of change</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="mean_distance_earth_r" id="mean_distance_earth_r"></td>
</tr>

<?
$er=-0.00003804;
$excentricity_earth_r=$er;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Excentricity er rate of
change</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="excentricity_earth_r" id="excentricity_earth_r"></td>
</tr>

<?
$ir=-46.94;
$inclin_orbit_earth_r=$ir;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Inclination ir rate of
change</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="inclin_orbit_earth_r" id="inclin_orbit_earth_r"></td>
</tr>

<?
$Wr=1198.28;
$long_perihelion_earth_r=$Wr;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Longitude of perihion Wr
rate of change</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="long_perihelion_earth_r" id="long_perihelion_earth_r"></td>
</tr>

<?
$Or=-18228.25;
$long_asc_node_earth_r=$Or;
?>
<tr>
```

```
<td valign=top height=20 colspan=1 width=300px>Earth Long acending node Or
rate of change</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="long_asc_node_earth_r" id="long_asc_node_earth_r"></td>
</tr>

<?
$lr=1293740.63;
$mean_long_earth_r=$lr;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Mean longitude lr rate of
change</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="mean_long_earth_r" id="mean_long_earth_r"></td>
</tr>

<?
$Nr=99;
$mean_long_earth_factor_r=$Nr;
?>
<tr>
<td valign=top height=20  colspan=1 width=300px>Earth Mean longitude Nr
factor</td>
<td valign=top height=20  colspan=2 width=200px><input type='text'
name="mean_long_earth_factor_r" id="mean_long_earth_factor_r"></td>
</tr>

   <tr><td width=500 valign=top height=10></td></tr>



<?
$t=$julian_century_time;
$at=$ao+$ar*$t;
$mean_distance_earth_t=$at;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Semi-major axis (at) at
time t</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="mean_distance_earth_t" id="mean_distance_earth_t"></td>
</tr>
```

```
<?

$et=$eo + $er*$t;
$excentricity_earth_t=$et;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Excentricity et at time
t</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="excentricity_earth_t" id="excentricity_earth_t"></td>
</tr>

<?
$Wt=$Wo + ($Wr / 3600)*$t;
$long_perihelion_earth_t=$Wt;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Longitude of perihion Wt
at time t</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="long_perihelion_earth_t" id="long_perihelion_earth_t"></td>
</tr>

<?
$Ot=$Oo + ($Or / 3600)*$t;
$long_asc_node_earth_t=$Ot;
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Long acending node Ot
at time t</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="long_asc_node_earth_t" id="long_asc_node_earth_t"></td>
</tr>

<?
$lt=$lo + ($lr / 3600 + 360*$Nr)*$t;
//$mean_long_earth_t=$lt;
$mean_long_earth_t=rev($lt);
?>
<tr>
<td valign=top height=20 colspan=1 width=300px>Earth Mean longitude lt at time
t</td>
<td valign=top height=20 colspan=2 width=200px><input type='text'
name="mean_long_earth_t" id="mean_long_earth_t"></td>
```

```
</tr>

<?
$lt=$mean_long_earth_t;
$Mt=$lt-$Wt;
$mean_anomaly_earth_t=$Mt;
?>
<tr>
<td valign=top height=20 colspan=1 width=400px>Earth Mean Anomaly Mt at
time t</td>
<td valign=top height=20 colspan=2><input type='text'
name="mean_anomaly_earth_t" id="mean_anomaly_earth_t"></td>
</tr>

<?
$E_tolerance=0.0000001;
$exc_anomaly_earth_t=ec_anomaly($mean_anomaly_earth_t,$excentricity_eart
h_t,$E_tolerance);
?>
<tr>
<td valign=top height=20 colspan=1 width=400px>Earth Exccentric Anomaly Et
at time t</td>
<td valign=top height=20 colspan=2><input type='text'
name="exc_anomaly_earth_t" id="exc_anomaly_earth_t"></td>
</tr>

<?
$Et=$exc_anomaly_earth_t;
$et=$excentricity_earth_t;
$vt=f_true_anomaly($Et,$et);
$true_anomaly_earth_t=$vt;
$lp=$heliocentric_long_proj;
?>
<tr>
<td valign=top height=20 colspan=1 width=400px>Earth True Anomaly vt at time
t</td>
<td valign=top height=20 colspan=2><input type='text'
name="true_anomaly_earth_t" id="true_anomaly_earth_t"></td>
</tr>

<?
$heliocentric_dist_earth_t=($at*(1-$et*$et))/(1+$et*COS(deg2rad($vt)));
?>
<tr>
```

```
<td valign=top height=20 colspan=1 width=400px>Earth Heliocentric distance rt
at time t</td>
<td valign=top height=20 colspan=2><input type='text'
name="heliocentric_dist_earth_t" id="heliocentric_dist_earth_t"></td>
</tr>



  <tr><td width=500 valign=top height=17></td></tr>

<?
 $le=$mean_long_earth_t;
 $rp=$heliocentric_dist_proj;
 $re=$heliocentric_dist_earth_t;
 $lp=$heliocentric_long_proj;

 //$rp=1.017551;
 //$lp=30.233224;

 $lambda=f_geo_ecliptic_long($le,$rp,$re,$lp);
 $geo_ecliptic_long=rev($lambda);
?>
<tr>
<td valign=top height=20 colspan=1 width=400px>Ast. Geocentric eclip. long.
(lambda) at time t</td>
<td valign=top height=20 colspan=2><input type='text' name="geo_ecliptic_long"
id="geo_ecliptic_long"></td>
</tr>

<?
 $le=$mean_long_earth_t;
 $rp=$heliocentric_dist_proj;
 $re=$heliocentric_dist_earth_t;
 $lp=$heliocentric_long_proj;
 $phi=$heliocentric_lat;

 //$rp=1.017551;
 //$lp=30.233224;
 //$phi=8.530351;

 $lambda=$geo_ecliptic_long;
 $beta=f_geo_ecliptic_lat($le,$rp,$re,$lp,$phi,$lambda);
 $geo_ecliptic_lat=rev($beta);
```

```
?>
<tr>
<td valign=top height=20 colspan=1 width=400px>Ast. Geocentric eclip. lat.
(beta) at time t</td>
<td valign=top height=20 colspan=2><input type='text' name="geo_ecliptic_lat"
id="geo_ecliptic_lat"></td>
</tr>

<?
  //=23.452294 - (  46.845*T_epsilon + 0.0059*POWER(T_epsilon,2) -
0.00181*POWER(T_epsilon,3)  ) / 3600
  $T_epsilon=$epsilon_century_time;
  $ecliptic_mean_obliquity=23.452294 - (  46.845*$T_epsilon +
0.0059*POW($T_epsilon,2) - 0.00181*POW($T_epsilon,3)  ) / 3600;
?>
<tr>
<td valign=top height=20 colspan=1 width=400px>Ast. Ecliptic Mean obliquity
(epsilon) at time t</td>
<td valign=top height=20 colspan=2><input type='text'
name="ecliptic_mean_obliquity" id="ecliptic_mean_obliquity"></td>
</tr>


   <tr><td width=500 valign=top height=24></td></tr>

<?

//=SIN(RADIANS(beta))*COS(RADIANS(epsilon))+COS(RADIANS(beta))*SIN(R
ADIANS(epsilon))*SIN(RADIANS(lambda))
  $pi=pi();
  $epsilon=$ecliptic_mean_obliquity;
  $beta=$geo_ecliptic_lat;
  $lambda=$geo_ecliptic_long;

$SIN_DELTA=SIN(deg2rad($beta))*COS(deg2rad($epsilon))+COS(deg2rad($be
ta))*SIN(deg2rad($epsilon))*SIN(deg2rad($lambda));
  $declination=180/$pi*ASIN($SIN_DELTA);
?>
<tr>
<td valign=top height=20 colspan=1 width=400px>Ast. Declination (delta) at time
t</td>
<td valign=top height=20 colspan=2><input type='text' name="declination"
id="declination"></td>
</tr>
```

```
<?
 $delta=$declination;
 $declination_degree=f_declination($delta);
?>
<tr>
<td valign=top height=20 colspan=1 width=400px>Ast. Declination
degs/mins/secs at time t</td>
<td valign=top height=20 colspan=2><input type='text'
name="declination_degree" id="declination_degree"></td>
</tr>


<?
  //SIN(RADIANS(lambda))*COS(RADIANS(epsilon)) -
TAN(RADIANS(beta))*SIN(RADIANS(epsilon))
  $pi=pi();
  $epsilon=$ecliptic_mean_obliquity;
  $beta=$geo_ecliptic_lat;
  $lambda=$geo_ecliptic_long;
  $alpha_y=SIN(deg2rad($lambda))*COS(deg2rad($epsilon)) -
TAN(deg2rad($beta))*SIN(deg2rad($epsilon));
  //=COS(RADIANS(lambda))
  $alpha_x=COS(deg2rad($lambda));
  //$right_ascention=f_right_ascension($lambda);
  $RA=180/$pi*ATAN2($alpha_y,$alpha_x);
  $right_ascention=rev($RA);
?>
<tr>
<td valign=top height=20  colspan=1 width=400px>Ast. right ascension RA
(alpha) at time t</td>
<td valign=top height=20  colspan=2><input type='text' name="right_ascention"
id="right_ascention"></td>
</tr>

<?
  $right_ascention_hours=f_right_ascention($right_ascention);
?>
<tr>
<td valign=top  colspan=1 width=400px>Ast. Right ascension RA in
hrs/mins/secs</td>
<td valign=top  colspan=2><input type='text' name="right_ascention_hours"
id="right_ascention_hours"></td>
```

```
</tr>
      </form>
</TABLE></td>


</TABLE>


  <?
    if ($calc_email_flag=='Y')
    {
      //$email_from="smallbodiesproject@gmail.com";
      $email_from="jeanmarcdenis@yahoo.com";
      $START_TIME='2008-Mar-04 05:39:59.97';
      $STOP_TIME='2008-Mar-05 05:39:59.97';

$email_result=email_to_horizon($email_from,$asteroidname,$START_TIME,$S
TOP_TIME);
      $calc_email_flag=='N';
    }
  ?>

  <?
    if ($match_email_flag=='Y')
    {
      $email_from="smallbodiesproject@gmail.com";
      //$email_from="jeanmarcdenis@yahoo.com";

      $select_match_max_cnt = "select count(AST_id) as cnt_max from
tbl_matches";
      $rs_max_cnt = mysql_query($select_match_max_cnt);
      while($row = mysql_fetch_array($rs_max_cnt))
      {
        $cnt_max=$row['cnt_max'];
      }

      //$cnt_max=3;

      for ($j=1;$j<=$cnt_max;$j++)
      {

       $select_astid = "SELECT
match_id,ast_name,date_format(PIC_DATE_OBS,'%Y-%b-%d')
PIC_DATE_OBS,date_format(PIC_DATE_OBS+1,'%Y-%b-%d')
```

```
PIC_DATE_OBS1,PIC_UT_STR from tbl_matches where HORIZONS_LINE IS
NULL and match_id=".$j."";
        $rs = mysql_query($select_astid);
        $k=0;
        while($row = mysql_fetch_array($rs))
        {
         $k++;
         $asteroidname=$row['ast_name'];
         $PIC_DATE_OBS=$row['PIC_DATE_OBS'];
         $PIC_DATE_OBS1=$row['PIC_DATE_OBS1'];
         $PIC_UT_STR=$row['PIC_UT_STR'];
         $START_TIME=$PIC_DATE_OBS." ".$PIC_UT_STR;
         $STOP_TIME=$PIC_DATE_OBS1." ".$PIC_UT_STR;

$email_result=email_to_horizon($email_from,$asteroidname,$START_TIME,$S
TOP_TIME);
        }
       }
       $match_email_flag=='N';
   }
  ?>

  <?
    if ($gen_matches_flag=='Y')
    {
       $email_from="smallbodiesproject@gmail.com";
       $qry="delete FROM tbl_matches";
       $rt=mysql_query($qry);
       $result = mysql_query( "CALL pic_FIND_MATCH()" );

       $gen_matches_flag=='N';
    }
  ?>

  <?
   if ($update_matches_flag=='Y')
   {
     $file = "windowtab.txt";


// First loop, old syle name like 'Patientia'

$fp = fopen($file, 'r');
//$i=0;
```

```
$read_next='false';

$select_min = "SELECT min(match_id) as minid from tbl_matches where
horizons_line is null;";
$rs = mysql_query($select_min);
  while($row = mysql_fetch_array($rs))
   {
     $min_id=$row['minid'];
   }
$i=$min_id-1;


while (!feof($fp))
{
    $sline = stream_get_line($fp, 10000, "\n");
    $i=$i+1;

    if (substr($sline,0,16)=='Target body name')
     {
     echo $sline;print "<br>";
     $pos=stripos($sline,"(");
     $pos2=stripos($sline,")");
     echo "pos= ".$pos;print "<br>";
     echo "pos2= ".$pos2;print "<br>";
     $len=strlen($sline);
     echo "len= ".$len;print "<br>";
     $name_pos=$len-$pos;
     $object_name=trim(substr($sline,($pos+1),($pos2-$pos-1)));print "<br>";
     echo "object name= ".$object_name;print "<br>";
     }

    if ($read_next=='true')
    {
     $PIC_TIME='';
     $UT_TIME=substr($sline,13,12);
     echo "UT time= ".$UT_TIME;print "<br>";
     $select_time = "SELECT PIC_UT_STR from tbl_matches where
AST_name='$object_name' and PIC_UT_STR='$UT_TIME' and horizons_line is
null;";


     //$select_time = "SELECT PIC_UT_STR from tbl_matches where
AST_name='$object_name';";
     $rs = mysql_query($select_time);
```

```
    if (!$rs) {echo $select_time;}
    while($row = mysql_fetch_array($rs))
    {
      $PIC_TIME=TRIM($row['PIC_UT_STR']);
    }
    echo "pictime=".$PIC_TIME;print "<br>";
    $read_next='false';
    echo $sline;print "<br>";
    $pos=stripos($sline,$PIC_TIME);
    $lentime=strlen($PIC_TIME);
    $len=strlen($sline);
    $name_pos_RA=$pos+$lentime;
    $RA=trim(substr($sline,$name_pos_RA,18));
    $name_pos_DEC=$pos+$lentime+18;
    $DEC=trim(substr($sline,$name_pos_DEC,15));
    //echo "RA= ".$RA;print "<br>";
    //echo "DEC= ".$DEC;print "<br>";
    $RA_VAL=convert_RA($RA);
    $DEC_VAL=convert_DEC($DEC);

    $MAG=trim(substr($sline,$name_pos_RA+53,8));
    //echo "MAG= ".$MAG;print "<br>";
    $PIC_TIME1=trim($PIC_TIME);
    if ($PIC_TIME1!='')
    {

$result=update_match($object_name,$RA,$DEC,$RA_VAL,$DEC_VAL,$MAG,$
sline,$PIC_TIME);
    }
   }

   if (substr($sline,0,5)=='$$SOE')
   {
    $read_next='true';
   }
   else
   {
    $read_next='false';
   }

}
fclose($fp);

// Second loop, new syle name like '1994 GV10'
```

```
$fp = fopen($file, 'r');
$read_next='false';


$select_min = "SELECT min(match_id) as minid from tbl_matches where
horizons_line is null;";
$rs = mysql_query($select_min);
  while($row = mysql_fetch_array($rs))
   {
     $min_id=$row['minid'];
   }
$i=$min_id-1;

while (!feof($fp))
{
    $sline = stream_get_line($fp, 10000, "\n");
    $i=$i+1;

    if (substr($sline,0,16)=='Target body name')
    //if (substr($sline,0,23)=='Target body name: 16874')
     {
     //$test_next='true';
     echo $sline;print "<br>";
     $sline1=substr($sline,18,strlen($sline)-18);
     echo $sline1;print "<br>";
     $pos=strpos($sline1," ");
     $pos2=strpos($sline1,"(");
     //echo "pos= ".$pos;print "<br>";
     //echo "pos2= ".$pos2;print "<br>";
     $len=strlen($sline1);
     //echo "len= ".$len;print "<br>";
     $name_pos=$len-$pos;
     $object_name=trim(substr($sline1,($pos+1),($pos2-$pos-1)));print "<br>";
     if ($object_name != '') {echo "object name= ".$object_name;print "<br>";}
     }

    if ($read_next=='true')
    {
     //$test_next='false';
     $PIC_TIME='';
     $UT_TIME=substr($sline,13,12);
     echo "UT time= ".$UT_TIME;print "<br>";
```

```
    $select_time = "SELECT PIC_UT_STR from tbl_matches where
AST_name='$object_name' and PIC_UT_STR='$UT_TIME' and horizons_line is
null;";
    $rs = mysql_query($select_time);
    if (!$rs) {echo $select_time;}
    while($row = mysql_fetch_array($rs))
    {
      $PIC_TIME=TRIM($row['PIC_UT_STR']);
    }
    //echo "pictime=".$PIC_TIME;print "<br>";
    $read_next='false';
    echo $sline;print "<br>";
    //$PIC_TIME="05:39:59.970";
    $pos=stripos($sline,$PIC_TIME);
    $lentime=strlen($PIC_TIME);
    $len=strlen($sline);
    $name_pos_RA=$pos+$lentime;
    $RA=trim(substr($sline,$name_pos_RA,18));
    $name_pos_DEC=$pos+$lentime+18;
    $DEC=trim(substr($sline,$name_pos_DEC,15));
    //echo "RA= ".$RA;print "<br>";
    //echo "DEC= ".$DEC;print "<br>";
    $RA_VAL=convert_RA($RA);
    $DEC_VAL=convert_DEC($DEC);

    $MAG=trim(substr($sline,$name_pos_RA+53,8));
    //echo "MAG= ".$MAG;print "<br>";
    $PIC_TIME1=trim($PIC_TIME);
    if ($PIC_TIME1!='')
    {
        $result=update_match($object_name,$RA,$DEC,$RA_VAL,$DEC_VAL,
        $MAG,$sline,$PIC_TIME);
    }
   }
   if (substr($sline,0,5)=='$$SOE')
   {
    $read_next='true';
   }
   else
   {
    $read_next='false';
   }

}
```

```
fclose($fp);
$update_matches_flag='N';
 }
 ?>

<script><? echo "document.getElementById('date_in').value
='".$date_in."';?></script>
<script><? echo "document.getElementById('hour_id').value
=".$hour_id.";?></script>
<script><? echo "document.getElementById('min_id').value
=".$min_id.";?></script>
<script><? echo "document.getElementById('sec_id').value
=".$sec_id.";?></script>
<script><? echo "document.getElementById('asteroidbyid').value
=".$asteroidbyid.";?></script>
<script><? echo "document.getElementById('asteroidbyname').value
='".$asteroidbyname."';?></script>
<script><? echo "document.getElementById('selectasteroid').value
='".$selectasteroid."';?></script>

<script><? echo "document.getElementById('asteroidid').value
='".$asteroidid."';?></script>
<script><? echo
"document.getElementById('asteroidid').disabled=true";?></script>
<script><? echo
"document.getElementById('asteroidid').style.background='lightgrey'";?></script>
<script><? echo
"document.getElementById('asteroidid').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('asteroidname').value
='".$asteroidname."';?></script>
<script><? echo
"document.getElementById('asteroidname').disabled=true";?></script>
<script><? echo
"document.getElementById('asteroidname').style.background='lightgrey'";?></scri
pt>
<script><? echo
"document.getElementById('asteroidname').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('excentricity').value
=".$excentricity.";?></script>
<script><? echo
"document.getElementById('excentricity').disabled=true";?></script>
```

```
<script><? echo
"document.getElementById('excentricity').style.background='lightgrey'";?></script
>
<script><? echo
"document.getElementById('excentricity').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('mean_distance').value
=".$mean_distance."";?></script>
<script><? echo
"document.getElementById('mean_distance').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_distance').style.background='lightgrey'";?></s
cript>
<script><? echo
"document.getElementById('mean_distance').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('arg_perihelion').value
=".$arg_perihelion."";?></script>
<script><? echo
"document.getElementById('arg_perihelion').disabled=true";?></script>
<script><? echo
"document.getElementById('arg_perihelion').style.background='lightgrey'";?></sc
ript>
<script><? echo
"document.getElementById('arg_perihelion').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('long_asc_node').value
=".$long_asc_node."";?></script>
<script><? echo
"document.getElementById('long_asc_node').disabled=true";?></script>
<script><? echo
"document.getElementById('long_asc_node').style.background='lightgrey'";?></s
cript>
<script><? echo
"document.getElementById('long_asc_node').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('inclin_of_orbit').value
=".$inclin_of_orbit."";?></script>
<script><? echo
"document.getElementById('inclin_of_orbit').disabled=true";?></script>
<script><? echo
"document.getElementById('inclin_of_orbit').style.background='lightgrey'";?></scr
ipt>
```

```
<script><? echo
"document.getElementById('inclin_of_orbit').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('inclin_orbit').value
=".$inclin_orbit."";?></script>
<script><? echo
"document.getElementById('inclin_orbit').disabled=true";?></script>
<script><? echo
"document.getElementById('inclin_orbit').style.background='lightgrey'";?></script
>
<script><? echo
"document.getElementById('inclin_orbit').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('long_perihelion').value
=".$long_perihelion."";?></script>
<script><? echo
"document.getElementById('long_perihelion').disabled=true";?></script>
<script><? echo
"document.getElementById('long_perihelion').style.background='lightgrey'";?></s
cript>
<script><?
echo"document.getElementById('long_perihelion').style.fontWeight='bold'";?></sc
ript>

<script><? echo "document.getElementById('epoch_osculation').value
='".$epoch_osculation."'";?></script>
<script><? echo
"document.getElementById('epoch_osculation').disabled=true";?></script>
<script><? echo
"document.getElementById('epoch_osculation').style.background='lightgrey'";?><
/script>
<script><? echo
"document.getElementById('epoch_osculation').style.fontWeight='bold'";?></scrip
t>

<script><? echo "document.getElementById('mean_anomaly1').value
=".$mean_anomaly1."";?></script>
<script><? echo
"document.getElementById('mean_anomaly1').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_anomaly1').style.background='lightgrey'";?></
script>
```

```
<script><? echo
"document.getElementById('mean_anomaly1').style.fontWeight='bold'";?></script
>

<script><? echo "document.getElementById('mean_anomaly2').value
=".$mean_anomaly2."";?></script>
<script><? echo
"document.getElementById('mean_anomaly2').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_anomaly2').style.background='lightgrey'";?></
script>
<script><? echo
"document.getElementById('mean_anomaly2').style.fontWeight='bold'";?></script
>

<script><? echo "document.getElementById('julian_days').value
=".$julian_days."";?></script>
<script><? echo
"document.getElementById('julian_days').disabled=true";?></script>
<script><? echo
"document.getElementById('julian_days').style.background='lightgrey'";?></script
>
<script><? echo
"document.getElementById('julian_days').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('epsilon_century_time').value
=".$epsilon_century_time."";?></script>
<script><? echo
"document.getElementById('epsilon_century_time').disabled=true";?></script>
<script><? echo
"document.getElementById('epsilon_century_time').style.background='lightgrey'";
?></script>
<script><? echo
"document.getElementById('epsilon_century_time').style.fontWeight='bold'";?></s
cript>

<script><? echo "document.getElementById('heliocentric_long').value
=".$heliocentric_long."";?></script>
<script><? echo
"document.getElementById('heliocentric_long').disabled=true";?></script>
<script><? echo
"document.getElementById('heliocentric_long').style.background='lightgrey'";?></
script>
```

```
<script><? echo
"document.getElementById('heliocentric_long').style.fontWeight='bold'";?></script
>

<script><? echo "document.getElementById('excentric_anomaly').value
=".$excentric_anomaly."";?></script>
<script><? echo
"document.getElementById('excentric_anomaly').disabled=true";?></script>
<script><? echo
"document.getElementById('excentric_anomaly').style.background='lightgrey'";?>
</script>
<script><? echo
"document.getElementById('excentric_anomaly').style.fontWeight='bold'";?></scri
pt>

<script><? echo "document.getElementById('true_anomaly').value
=".$true_anomaly."";?></script>
<script><? echo
"document.getElementById('true_anomaly').disabled=true";?></script>
<script><? echo
"document.getElementById('true_anomaly').style.background='lightgrey'";?></scri
pt>
<script><? echo
"document.getElementById('true_anomaly').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('heliocentric_dist').value
=".$heliocentric_dist."";?></script>
<script><? echo
"document.getElementById('heliocentric_dist').disabled=true";?></script>
<script><? echo
"document.getElementById('heliocentric_dist').style.background='lightgrey'";?></
script>
<script><? echo
"document.getElementById('heliocentric_dist').style.fontWeight='bold'";?></script
>

<script><? echo "document.getElementById('heliocentric_lat').value
=".$heliocentric_lat."";?></script>
<script><? echo
"document.getElementById('heliocentric_lat').disabled=true";?></script>
<script><? echo
"document.getElementById('heliocentric_lat').style.background='lightgrey'";?></s
cript>
```

```
<script><? echo
"document.getElementById('heliocentric_lat').style.fontWeight='bold'";?></script>

<script><? echo "document.getElementById('x_cart_ecliptic_coord').value
=".$x_cart_ecliptic_coord."";?></script>
<script><? echo
"document.getElementById('x_cart_ecliptic_coord').disabled=true";?></script>
<script><? echo
"document.getElementById('x_cart_ecliptic_coord').style.background='lightgrey'";
?></script>
<script><? echo
"document.getElementById('x_cart_ecliptic_coord').style.fontWeight='bold'";?></s
cript>

<script><? echo "document.getElementById('y_cart_ecliptic_coord').value
=".$y_cart_ecliptic_coord."";?></script>
<script><? echo
"document.getElementById('y_cart_ecliptic_coord').disabled=true";?></script>
<script><? echo
"document.getElementById('y_cart_ecliptic_coord').style.background='lightgrey'";
?></script>
<script><? echo
"document.getElementById('y_cart_ecliptic_coord').style.fontWeight='bold'";?></s
cript>

<script><? echo "document.getElementById('polar_ecliptic_angle').value
=".$polar_ecliptic_angle."";?></script>
<script><? echo
"document.getElementById('polar_ecliptic_angle').disabled=true";?></script>
<script><? echo
"document.getElementById('polar_ecliptic_angle').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('polar_ecliptic_angle').style.fontWeight='bold'";?></sc
ript>

<script><? echo "document.getElementById('heliocentric_long_proj').value
=".$heliocentric_long_proj."";?></script>
<script><? echo
"document.getElementById('heliocentric_long_proj').disabled=true";?></script>
<script><? echo
"document.getElementById('heliocentric_long_proj').style.background='lightgrey'"
;?></script>
```

```
<script><? echo
"document.getElementById('heliocentric_long_proj').style.fontWeight='bold'";?></
script>

<script><? echo "document.getElementById('heliocentric_dist_proj').value
=".$heliocentric_dist_proj."";?></script>
<script><? echo
"document.getElementById('heliocentric_dist_proj').disabled=true";?></script>
<script><? echo
"document.getElementById('heliocentric_dist_proj').style.background='lightgrey'";
?></script>
<script><? echo
"document.getElementById('heliocentric_dist_proj').style.fontWeight='bold'";?></s
cript>


<script><? echo "document.getElementById('mean_distance_earth_o').value
=".$mean_distance_earth_o."";?></script>
<script><? echo
"document.getElementById('mean_distance_earth_o').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_distance_earth_o').style.background='lightgre
y'";?></script>
<script><? echo
"document.getElementById('mean_distance_earth_o').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('excentricity_earth_o').value
=".$excentricity_earth_o."";?></script>
<script><? echo
"document.getElementById('excentricity_earth_o').disabled=true";?></script>
<script><? echo
"document.getElementById('excentricity_earth_o').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('excentricity_earth_o').style.fontWeight='bold'";?></sc
ript>

<script><? echo "document.getElementById('inclin_orbit_earth_o').value
=".$inclin_orbit_earth_o."";?></script>
<script><? echo
"document.getElementById('inclin_orbit_earth_o').disabled=true";?></script>
```

```
<script><? echo
"document.getElementById('inclin_orbit_earth_o').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('inclin_orbit_earth_o').style.fontWeight='bold'";?></sc
ript>

<script><? echo "document.getElementById('long_perihelion_earth_o').value
=".$long_perihelion_earth_o."";?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_o').disabled=true";?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_o').style.background='lightgre
y'";?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_o').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('long_asc_node_earth_o').value
=".$long_asc_node_earth_o."";?></script>
<script><? echo
"document.getElementById('long_asc_node_earth_o').disabled=true";?></script>
<script><? echo
"document.getElementById('long_asc_node_earth_o').style.background='lightgre
y'";?></script>
<script><? echo
"document.getElementById('long_asc_node_earth_o').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('mean_long_earth_o').value
=".$mean_long_earth_o."";?></script>
<script><? echo
"document.getElementById('mean_long_earth_o').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_long_earth_o').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('mean_long_earth_o').style.fontWeight='bold'";?></sc
ript>


<script><? echo "document.getElementById('mean_distance_earth_r').value
=".$mean_distance_earth_r."";?></script>
```

```
<script><? echo
"document.getElementById('mean_distance_earth_r').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_distance_earth_r').style.background='lightgre
y'";?></script>
<script><? echo
"document.getElementById('mean_distance_earth_r').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('excentricity_earth_r').value
=".$excentricity_earth_r."";?></script>
<script><? echo
"document.getElementById('excentricity_earth_r').disabled=true";?></script>
<script><? echo
"document.getElementById('excentricity_earth_r').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('excentricity_earth_r').style.fontWeight='bold'";?></sc
ript>

<script><? echo "document.getElementById('inclin_orbit_earth_r').value
=".$inclin_orbit_earth_r."";?></script>
<script><? echo
"document.getElementById('inclin_orbit_earth_r').disabled=true";?></script>
<script><? echo
"document.getElementById('inclin_orbit_earth_r').style.background='lightgrey'";?>
</script>
<script><? echo
"document.getElementById('inclin_orbit_earth_r').style.fontWeight='bold'";?></scr
ipt>

<script><? echo "document.getElementById('long_perihelion_earth_r').value
=".$long_perihelion_earth_r."";?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_r').disabled=true";?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_r').style.background='lightgre
y'";?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_r').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('long_asc_node_earth_r').value
=".$long_asc_node_earth_r."";?></script>
```

```
<script><? echo
"document.getElementById('long_asc_node_earth_r').disabled=true";?></script>
<script><? echo
"document.getElementById('long_asc_node_earth_r').style.background='lightgre
y'";?></script>
<script><? echo
"document.getElementById('long_asc_node_earth_r').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('mean_long_earth_r').value
=".$mean_long_earth_r."";?></script>
<script><? echo
"document.getElementById('mean_long_earth_r').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_long_earth_r').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('mean_long_earth_r').style.fontWeight='bold'";?></sc
ript>

<script><? echo "document.getElementById('mean_long_earth_factor_r').value
=".$mean_long_earth_factor_r."";?></script>
<script><? echo
"document.getElementById('mean_long_earth_factor_r').disabled=true";?></scrip
t>
<script><? echo
"document.getElementById('mean_long_earth_factor_r').style.background='lightg
rey'";?></script>
<script><? echo
"document.getElementById('mean_long_earth_factor_r').style.fontWeight='bold'";
?></script>

<script><? echo "document.getElementById('julian_century_time').value
=".$julian_century_time."";?></script>
<script><? echo
"document.getElementById('julian_century_time').disabled=true";?></script>
<script><? echo
"document.getElementById('julian_century_time').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('julian_century_time').style.fontWeight='bold'";?></scr
ipt>
```

```
<script><? echo "document.getElementById('mean_distance_earth_t').value
=".$mean_distance_earth_t."";?></script>
<script><? echo
"document.getElementById('mean_distance_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_distance_earth_t').style.background='lightgrey
';?></script>
<script><? echo
"document.getElementById('mean_distance_earth_t').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('excentricity_earth_t').value
=".$excentricity_earth_t."";?></script>
<script><? echo
"document.getElementById('excentricity_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('excentricity_earth_t').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('excentricity_earth_t').style.fontWeight='bold'";?></sc
ript>

<script><? echo "document.getElementById('long_perihelion_earth_t').value
=".$long_perihelion_earth_t."";?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_t').style.background='lightgrey
';?></script>
<script><? echo
"document.getElementById('long_perihelion_earth_t').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('long_asc_node_earth_t').value
=".$long_asc_node_earth_t."";?></script>
<script><? echo
"document.getElementById('long_asc_node_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('long_asc_node_earth_t').style.background='lightgrey
';?></script>
<script><? echo
"document.getElementById('long_asc_node_earth_t').style.fontWeight='bold'";?>
</script>
```

```
<script><? echo "document.getElementById('mean_long_earth_t').value
=".$mean_long_earth_t."";?></script>
<script><? echo
"document.getElementById('mean_long_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_long_earth_t').style.background='lightgrey'";?
></script>
<script><? echo
"document.getElementById('mean_long_earth_t').style.fontWeight='bold'";?></scr
ipt>

<script><? echo "document.getElementById('mean_long_earth_factor_t').value
=".$mean_long_earth_factor_t."";?></script>
<script><? echo
"document.getElementById('mean_long_earth_factor_t').disabled=true";?></scrip
t>
<script><? echo
"document.getElementById('mean_long_earth_factor_t').style.background='lightg
rey'";?></script>
<script><? echo
"document.getElementById('mean_long_earth_factor_t').style.fontWeight='bold'";
?></script>

<script><? echo "document.getElementById('mean_anomaly_earth_t').value
=".$mean_anomaly_earth_t."";?></script>
<script><? echo
"document.getElementById('mean_anomaly_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('mean_anomaly_earth_t').style.background='lightgre
y'";?></script>
<script><? echo
"document.getElementById('mean_anomaly_earth_t').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('exc_anomaly_earth_t').value
=".$exc_anomaly_earth_t."";?></script>
<script><? echo
"document.getElementById('exc_anomaly_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('exc_anomaly_earth_t').style.background='lightgrey'";
?></script>
<script><? echo
"document.getElementById('exc_anomaly_earth_t').style.fontWeight='bold'";?></
script>
```

```
<script><? echo "document.getElementById('true_anomaly_earth_t').value
=".$true_anomaly_earth_t."";?></script>
<script><? echo
"document.getElementById('true_anomaly_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('true_anomaly_earth_t').style.background='lightgrey'"
;?></script>
<script><? echo
"document.getElementById('true_anomaly_earth_t').style.fontWeight='bold'";?></
script>

<script><? echo "document.getElementById('heliocentric_dist_earth_t').value
=".$heliocentric_dist_earth_t."";?></script>
<script><? echo
"document.getElementById('heliocentric_dist_earth_t').disabled=true";?></script>
<script><? echo
"document.getElementById('heliocentric_dist_earth_t').style.background='lightgre
y'";?></script>
<script><? echo
"document.getElementById('heliocentric_dist_earth_t').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('geo_ecliptic_long').value
=".$geo_ecliptic_long."";?></script>
<script><? echo
"document.getElementById('geo_ecliptic_long').disabled=true";?></script>
<script><? echo
"document.getElementById('geo_ecliptic_long').style.background='lightgrey'";?><
/script>
<script><? echo
"document.getElementById('geo_ecliptic_long').style.fontWeight='bold'";?></scrip
t>

<script><? echo "document.getElementById('geo_ecliptic_lat').value
=".$geo_ecliptic_lat."";?></script>
<script><? echo
"document.getElementById('geo_ecliptic_lat').disabled=true";?></script>
<script><? echo
"document.getElementById('geo_ecliptic_lat').style.background='lightgrey'";?></s
cript>
<script><? echo
"document.getElementById('geo_ecliptic_lat').style.fontWeight='bold'";?></script>
```

```
<script><? echo "document.getElementById('ecliptic_mean_obliquity').value
=".$ecliptic_mean_obliquity."";?></script>
<script><? echo
"document.getElementById('ecliptic_mean_obliquity').disabled=true";?></script>
<script><? echo
"document.getElementById('ecliptic_mean_obliquity').style.background='lightgrey
';?></script>
<script><? echo
"document.getElementById('ecliptic_mean_obliquity').style.fontWeight='bold'";?>
</script>

<script><? echo "document.getElementById('declination').value
=".$declination."";?></script>
<script><? echo
"document.getElementById('declination').disabled=true";?></script>
<script><? echo
"document.getElementById('declination').style.background='lightgrey'";?></script
>
<script><? echo
"document.getElementById('declination').style.fontWeight='bold'";?></script>


<script><? echo "document.getElementById('declination_degree').value
='".$declination_degree."';?></script>
<script><? echo
"document.getElementById('declination_degree').disabled=true";?></script>
<script><? echo
"document.getElementById('declination_degree').style.background='lightgrey'";?>
</script>
<script><? echo
"document.getElementById('declination_degree').style.fontWeight='bold'";?></scr
ipt>

<script><? echo "document.getElementById('right_ascention').value
=".$right_ascention."";?></script>
<script><? echo
"document.getElementById('right_ascention').disabled=true";?></script>
<script><? echo
"document.getElementById('right_ascention').style.background='lightgrey'";?></s
cript>
<script><? echo
"document.getElementById('right_ascention').style.fontWeight='bold'";?></script>
```

```
<script><? echo "document.getElementById('right_ascention_hours').value
='".$right_ascention_hours."';?></script>
<script><? echo
"document.getElementById('right_ascention_hours').disabled=true";?></script>
<script><? echo
"document.getElementById('right_ascention_hours').style.background='lightgrey'"
;?></script>
<script><? echo
"document.getElementById('right_ascention_hours').style.fontWeight='bold'";?></
script>

<script><? echo "document.getElementById('calc_procedure_flag').value
='".$calc_procedure_flag."';?></script>

<script>
setWidth()
</script>

</body>
</html>
```

APPENDIX E: LIST OF IDENTIFIED ASTEROIDS

* Diameter given, not calculated from absolute magnitude
# Blank average offsets are from asteroids in non-calibrated picture frames

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)		Avg Offset (arcsec)
1	1036 T-2	24578	14.6	5.713	*	0.26
2	1176 T-2	17328	15.3	2.841	*	0.40
3	1964 TT2	4047	13.1	10.829	*	0.67
4	1985 CV1	18326	13.6	8.902	*	0.70
5	1990 QT9	10312	14.9	3.593		1.51
6	1992 SB3	150115	16.1	2.068		2.35
7	1993 GL	11568	12.6	10.363		0.78
8	1993 TN2	13582	13.3	7.507		1.87
9	1994 AY8	173126	16.2	1.975		0.22
10	1994 SQ11	230421	17.1	1.305		0.53
11	1995 FA14	219035	16.2	1.975		0.29
12	1995 SE	136666	16.7	1.568		0.62
13	1995 UE39	200108	16.7	1.568		0.77
14	1995 YS21	22424	13.3	4.134	*	1.15
15	1996 AX8	100409	16.2	1.975		1.70
16	1996 BV7	231536	16.2	1.975		0.34
17	1996 TA17	231758	16.27	1.912		
18	1996 XS13	69477	16.8	1.498		0.71
19	1997 AE22	69496	16.3	1.886		1.26
20	1997 AY18	100511	15	3.431		
21	1997 EP4	190304	16.4	1.801		1.92
22	1997 ES12	232097	17.79	0.949		1.05
23	1997 GP34	29432	14.4	4.524		0.87
24	1997 NS7	192417	16.2	1.975		0.27
25	1997 PC5	52590	15.7	2.486		0.16
26	1997 WM5	85523	15.2	3.13		1.96
27	1997 WZ27	85535	15.3	2.989		0.48
28	1997 YM5	33095	14.8	2.998	*	0.39
29	1998 BQ20	192455	15.5	2.726		0.69
30	1998 BT36	69575	17.3	1.19		1.19
31	1998 DU20	35460	15	3.431		2.60
32	1998 OD12	22649	14.1	5.194		0.63
33	1998 OP12	33320	14.7	3.94		1.36
34	1998 QV37	22672	13.8	5.963		1.45
35	1998 SB72	23906	14.4	8.773	*	1.81
36	1998 SF9	58993	14.6	6.698	*	1.03
37	1998 TN4	101340	15.7	2.486		
38	1998 VN46	233655	18.23	0.775		0.18
39	1998 XJ24	79852	16	2.165		0.50
40	1999 BV19	31438	15.1	1.784	*	0.75
41	1999 CC137	59346	16.3	2.489	*	0.48
42	1999 CC38	44559	13.9	7.795	*	0.53
43	1999 CK71	28302	13.8	10.514	*	0.89

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)		Avg Offset (arcsec)
44	1999 JG82	80060	14.5	8.552	*	0.98
45	1999 JS71	33627	13.5	8.991	*	0.57
46	1999 JX121	59707	15.2	3.13		
47	1999 RC85	189459	16.7	1.568		1.43
48	1999 RE55	28398	15.5	2.726		1.11
49	1999 RQ128	234659	16.64	1.612		63.36
50	1999 RU200	96836	14.9	3.593		0.65
51	1999 TB124	192713	15.9	2.267		1.44
52	1999 TB195	129947	16.2	1.975		1.34
53	1999 TV23	137319	16.8	1.498		1.11
54	1999 TV256	121491	15.1	3.277		0.36
55	1999 TY272	148135	16.2	1.975		0.89
56	1999 TZ140	40895	15.1	3.277		0.46
57	1999 TZ283	102501	16.2	1.975		1.09
58	1999 VQ173	75197	16	2.165		0.87
59	1999 VS29	134650	16.2	1.975		1.08
60	1999 VX210	70927	15.2	3.13		1.94
61	1999 XC11	162315	15.9	2.267		0.85
62	1999 XD172	162347	15.4	2.854		0.70
63	1999 XP31	21996	12.4	6.807	*	0.36
64	1999 XW45	137680	16.8	1.498		1.02
65	1999 YX10	60324	16.4	1.801		3.45
66	2000 AJ104	75693	16	2.165		0.51
67	2000 AT225	137905	16.3	1.886		0.48
68	2000 AV246	18888	13.6	6.539		1.21
69	2000 BX27	50264	14	8.822	*	1.37
70	2000 CB91	71532	14.1	8.74	*	1.07
71	2000 CC34	16205	15.2	2.663	*	2.47
72	2000 CL16	80706	15.6	2.603		2.50
73	2000 CT79	86474	14.9	2.88	*	0.44
74	2000 DF41	50451	16	2.165		0.25
75	2000 DG18	138037	15.6	2.603		0.69
76	2000 DL94	47755	15.1	3.277		0.24
77	2000 EC36	121947	17	1.366		0.36
78	2000 EP56	17238	14	8.165	*	0.23
79	2000 ET30	53724	16.2	1.975		0.60
80	2000 EX60	60538	14.9	3.593		0.46
81	2000 EZ141	15149	13.1	8.232		1.79
82	2000 FB24	76432	13.3	7.507		
83	2000 GE6	104448	15.3	3.593		3.20
84	2000 GJ131	239029	18.54	0.672		0.68
85	2000 GR52	81354	15.1	4.863	*	0.60
86	2000 GV76	30177	14.4	4.524		0.32

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)		Avg Offset (arcsec)
87	2000 GY27	56429	16.5	1.72		0.18
88	2000 HE10	28758	13.5	10.503	*	0.72
89	2000 HE9	86856	16.5	1.72		0.65
90	2000 HR13	118620	16	2.165		
91	2000 HX15	54115	15.7	2.486		0.31
92	2000 JU14	81693	14.3	9.068	*	0.77
93	2000 KH18	28875	12.7	12.112	*	1.23
94	2000 KO55	54375	13.3	6.24	*	0.93
95	2000 KW3	87036	16	2.165		0.25
96	2000 LD20	87107	15	3.431		0.17
97	2000 LH27	33927	15.7	1.946	*	0.78
98	2000 OJ58	130416	16.2	1.975		1.03
99	2000 OS58	32261	14.9	3.593		0.47
100	2000 QH202	168758	16.9	1.43		0.91
101	2000 QL64	41487	15.4	2.854		1.32
102	2000 QR138	130502	16.9	1.43		
103	2000 SH220	165260	16.1	2.068		
104	2000 SM73	182152	16.5	1.72		1.32
105	2000 ST139	241209	16.35	1.843		
106	2000 SX184	62424	14.2	8.991	*	0.49
107	2000 TV35	242212	16.14	2.03		0.06
108	2000 TY66	165300	16.8	1.498		0.31
109	2000 TZ50	123164	16.1	2.068		1.31
110	2000 UN53	62828	14.3	7.387	*	0.85
111	2000 VB61	242840	15.45	2.789		0.71
112	2000 VC42	106422	17	1.366		0.64
113	2000 WV187	165388	16.4	1.801		0.76
114	2000 XF2	131005	14.8	3.763		1.17
115	2000 XZ11	93982	13.7	8.896	*	0.54
116	2000 YM4	94049	15.6	2.603		0.13
117	2000 YW70	200493	16.1	2.068		
118	2001 AV10	43465	15.3	2.989		
119	2001 BE52	131139	15.6	2.603		0.72
120	2001 BM33	24511	15.4	2.854		1.13
121	2001 BR37	134940	16.6	1.642		0.43
122	2001 BR62	94229	14.4	3.071	*	0.74
123	2001 CU42	107387	16.9	1.43		0.78
124	2001 DG10	244352	16.59	1.65		
125	2001 DR10	94268	14.6	4.963	*	1.36
126	2001 FF80	46165	14.5	5.816	*	1.29
127	2001 FG122	28964	13.9	9.63		0.77
128	2001 FL26	99084	14.8	5.071		1.02
129	2001 FM49	99100	15.1	3.277		1.17

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)		Avg Offset (arcsec)
130	2001 FV81	99126	14.4	4.225		0.29
131	2001 GW4	108121	14.9	3.593		1.63
132	2001 KV39	51725	14.2	6.836	*	1.34
133	2001 LK19	28983	15.4	2.854		0.35
134	2001 LZ9	108545	15.9	2.267		0.38
135	2001 MV27	54813	17.1	1.305		0.34
136	2001 OF11	245508	17.2	1.246		0.68
137	2001 OR72	88344	16.6	1.642		0.39
138	2001 PG4	146359	15.7	2.486		0.57
139	2001 PT3	108896	15.6	2.603		0.27
140	2001 QV173	246737	14.88	3.626		0.75
141	2001 QX157	193873	16.8	1.498		0.20
142	2001 RJ34	247611	16.72	1.554		0.51
143	2001 RJ90	190805	16.5	1.72		0.39
144	2001 RT30	247594	15.57	2.639		0.97
145	2001 RU131	83333	16.5	1.72		0.49
146	2001 SA325	250358	17.44	1.116		0.33
147	2001 SC91	248769	16.18	1.993		0.74
148	2001 SD23	190824	16.7	1.568		0.53
149	2001 SF172	64032	17.6	1.036		0.28
150	2001 SG176	249367	16.4	1.801		0.29
151	2001 SJ1	182436	16.4	1.801		0.59
152	2001 SS265	55382	15	3.518	*	0.43
153	2001 SS283	153602	17.7	0.99		0.26
154	2001 SW84	194037	17.2	1.246		0.44
155	2001 TC175	251674	17.49	1.09		0.49
156	2001 TD128	110593	15.5	2.726		0.38
157	2001 TF227	125055	16.1	2.068		0.51
158	2001 TM71	250991	16.6	1.642		0.50
159	2001 TW9	194153	16.9	1.43		
160	2001 TY126	158257	16.9	1.43		0.16
161	2001 TY50	119401	16.9	1.43		0.43
162	2001 UB178	189723	15.8	2.374		0.66
163	2001 UV127	110914	14.8	3.763		0.57
164	2001 UX169	156244	16.8	1.498		0.28
165	2001 VJ40	89315	14.9	2.782	*	0.32
166	2001 VM34	37373	14.2	4.427	*	0.36
167	2001 VO112	254573	17.54	1.065		4.97
168	2001 WV25	57815	15.2	3.13		
169	2001 WX52	131591	17.3	1.19		0.77
170	2001 YH11	99374	16.4	1.801		0.54
171	2001 YX94	131731	16.6	1.642		0.71
172	2002 AE117	39348	14.6	6.958	*	0.26

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)		Avg Offset (arcsec)
173	2002 AM166	89809	14.6	4.126		0.61
174	2002 AV7	126170	16	2.165		0.45
175	2002 CB82	126525	15.4	2.854		0.32
176	2002 CQ214	203578	17.3	1.19		0.58
177	2002 CT149	65158	16.6	1.642		0.63
178	2002 CT280	260379	17.46	1.105		0.27
179	2002 CY167	182862	15.4	2.854		0.44
180	2002 EA26	99513	15.8	1.608	*	1.41
181	2002 EB93	95551	14.9	3.593		0.25
182	2002 EQ48	261058	17.56	1.056		0.38
183	2002 EU21	111839	15.4	2.854		1.33
184	2002 EW128	261497	15.52	2.701		0.56
185	2002 GB41	188154	16.3	1.886		0.28
186	2002 GL26	73107	15.3	2.989		0.28
187	2002 GT59	160224	15	3.431		0.30
188	2002 GW69	166263	16.7	1.568		0.45
189	2002 HH12	112037	16.6	1.642		0.86
190	2002 NH16	112358	16.3	1.886		2.96
191	2002 NP56	174142	15.9	2.267		0.49
192	2002 NY3	218100	17.6	1.036		1.18
193	2002 OL30	195683	15.8	2.374		0.09
194	2002 PF40	78282	15.6	2.603		0.25
195	2002 PM159	266114	17.49	1.09		0.23
196	2002 PZ180	169798	16	2.165		0.99
197	2002 QF107	220031	17.7	0.99		
198	2002 QG105	268016	16.86	1.457		0.67
199	2002 QJ24	266887	15.49	2.738		0.40
200	2002 QV123	211254	16.5	1.72		0.27
201	2002 RB44	113022	15.4	2.854		0.34
202	2002 RD248	270480	16.89	1.437		
203	2002 RR46	78471	15.1	3.277		
204	2002 RT9	142126	17.2	1.246		0.62
205	2002 RX13	112958	16.4	1.886		0.37
206	2002 RZ203	270050	16.04	2.126		1.16
207	2002 SD5	271113	16.28	1.903		0.58
208	2002 SN27	147078	16.5	1.72		0.61
209	2002 SO63	271572	17.88	0.911		2.96
210	2002 SX4	203774	15.8	2.374		0.96
211	2002 TK134	272739	17.24	1.223		0.15
212	2002 TQ136	142612	15.4	2.854		0.75
213	2002 TQ251	273492	17.21	1.24		0.16
214	2002 TS248	213693	16.2	1.975		0.82
215	2002 TV66	89993	16	2.165		0.38

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)		Avg Offset (arcsec)
216	2002 TZ349	274160	16.53	1.696		5.91
217	2002 UT23	142800	16.5	1.72		
218	2002 UT46	142818	15.5	2.726		1.50
219	2002 VM26	275261	16.83	1.477		0.61
220	2002 VO32	172335	15.3	2.989		0.27
221	2002 VU72	215525	16.9	1.366		0.68
222	2002 XG64	156740	16.3	1.886		0.38
223	2002 XH31	143122	15.4	2.854		0.33
224	2002 XO75	174450	15.5	2.854		2.89
225	2003 AN39	196192	16.5	1.72		0.12
226	2003 CA18	127648	17	1.366		0.69
227	2003 EO36	196284	17.2	1.246		0.39
228	2003 FA99	279964	17.07	1.323		0.49
229	2003 FX103	156935	16.6	1.642		0.57
230	2003 FZ129	280107	7.38	114.678		0.42
231	2003 FZ83	279894	17.15	1.275		0.78
232	2003 GZ47	120145	14.1	5.194		0.43
233	2003 HC39	133072	16.6	1.642		
234	2003 NE13	281388	17.61	1.032		0.71
235	2003 OY15	196526	15.5	2.726		1.10
236	2003 OY27	163765	16.4	1.801		1.05
237	2003 QN32	170239	16.9	1.43		1.61
238	2003 QQ110	201548	15.9	2.267		1.04
239	2003 QQ28	282022	16.41	1.793		2.71
240	2003 QS58	282268	16.28	1.903		1.44
241	2003 RE17	282918	16.43	1.776		0.33
242	2003 SA266	285760	17.17	1.263		1.40
243	2003 SD394	287045	17.17	1.263		
244	2003 SE324	286407	18.33	0.74		0.99
245	2003 SE94	284053	15.81	2.363		0.81
246	2003 SH1	196632	16.6	1.642		0.59
247	2003 SM27	283308	15.96	2.205		
248	2003 SN61	283682	17.03	1.347		0.81
249	2003 SP152	203967	15.9	2.267		0.20
250	2003 SS11	209080	17	1.366		0.35
251	2003 SU318	286308	16.59	1.65		0.63
252	2003 SX40	283469	15.49	2.738		0.53
253	2003 SZ53	157045	15.1	3.277		0.74
254	2003 UA213	167298	15.5	2.726		0.76
255	2003 UD18	288069	16.5	1.72		1.29
256	2003 UD82	288701	17.28	1.201		0.45
257	2003 UG226	84913	14	7.086	*	0.20
258	2003 UK317	290774	17.55	1.06		7.33

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)	Avg Offset (arcsec)
259	2003 UM323	290809	17.24	1.223	3.73
260	2003 UV309	197185	15.8	2.374	
261	2003 UW196	177221	16.6	1.642	0.88
262	2003 UY	206468	16.8	1.498	0.48
263	2003 UZ110	288958	17.42	1.126	0.67
264	2003 WL43	291746	17.98	0.87	0.37
265	2003 WR69	191560	16	2.165	0.19
266	2003 WX117	292422	17.22	1.234	0.55
267	2003 WZ35	143767	17.2	1.246	0.99
268	2003 YA128	177343	16.2	1.975	0.61
269	2003 YG9	293364	16.59	1.65	0.33
270	2003 YJ88	197440	14.8	3.763	0.92
271	2003 YL179	294558	7.59	104.107	1.96
272	2003 YM174	170674	15.6	2.603	
273	2003 YM33	213963	16.2	1.975	0.51
274	2003 YN56	183709	16.5	1.72	0.52
275	2003 YQ17	293446	15.14	3.217	0.56
276	2003 YT49	170615	16	2.165	0.76
277	2003 YU166	116441	16.5	1.72	0.20
278	2003 YW179	294568	6.83	147.734	0.84
279	2003 YY100	294026	16.7	1.568	1.63
280	2004 BR147	170750	15.8	2.374	0.30
281	2004 CN2	295900	15.98	2.185	0.59
282	2004 DZ56	296998	17.28	1.201	0.56
283	2004 EL76	297760	15.87	2.299	0.59
284	2004 EW49	297562	16.62	1.568	0.79
285	2004 FP23	298126	16.74	1.498	0.67
286	2004 FZ91	116862	16.2	1.975	0.60
287	2004 HQ36	197591	16.7	1.568	0.54
288	2004 HZ24	299563	16.68	1.583	0.96
289	2004 PC14	180762	15.5	2.726	0.47
290	2004 QD9	133889	15.4	2.854	0.37
291	2004 QX19	157195	15.3	2.989	0.35
292	2004 RD228	204333	16.8	1.498	0.60
293	2004 RD341	306052	15.1	3.277	0.39
294	2004 RF7	128626	14.3	4.737	0.97
295	2004 RO303	128820	16.4	1.801	1.51
296	2004 RS186	198005	16	2.165	
297	2004 RX221	186976	15.3	2.989	0.35
298	2004 SQ60	161523	15	3.431	1.47
299	2004 TA288	309780	17.33	1.173	
300	2004 TJ70	128925	15.2	3.13	
301	2004 TK90	307862	17.65	1.013	

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)	Avg Offset (arcsec)
302	2004 XB61	311996	15.67	2.52	
303	2004 XC41	170906	17	1.366	0.58
304	2004 XN145	312657	14.92	3.56	0.57
305	2004 XQ153	198562	16.8	1.498	0.34
306	2004 XT32	229229	15.7	2.486	0.51
307	2004 YE4	312868	15.92	2.246	0.85
308	2005 AH54	164328	15.8	2.374	0.21
309	2005 AS29	313399	17.12	1.293	13.25
310	2005 BJ12	313995	17.33	1.173	0.66
311	2005 EE327	317890	16.18	1.993	0.52
312	2005 EP154	212103	15.5	2.726	0.13
313	2005 FY6	136470	15.2	3.13	0.61
314	2005 GC86	209815	16	2.165	0.40
315	2005 GP140	319321	16.02	2.145	0.36
316	2005 GQ51	172900	16.2	1.975	0.33
317	2005 GV29	218563	16.9	1.43	0.54
318	2005 JD87	320641	16.15	2.021	0.41
319	2005 JG132	136502	15	3.431	
320	2005 JS87	320645	16.85	1.464	
321	2005 JX178	321264	17.35	1.163	1.19
322	2005 LT49	177800	17.3	1.19	0.60
323	2005 NR56	323039	17.24	1.223	0.59
324	2005 QC92	325283	16.17	2.002	0.56
325	2005 QF4	324156	15.89	2.165	0.07
326	2005 SD167	329733	15.53	2.374	0.38
327	2005 SM198	181215	15.9	2.267	0.49
328	2005 TR194	333636	16.31	1.877	0.22
329	2005 TV45	187274	16	2.165	0.15
330	2005 UW511	339903	16.09	2.077	0.29
331	2005 VD17	340479	15.31	2.975	0.20
332	2005 VY123	341416	16.47	1.744	0.50
333	2005 WA22	342018	16.88	1.444	0.30
334	2005 WB22	227437	16.5	1.72	0.22
335	2005 WF152	343752	16.71	1.561	0.44
336	2005 WG13	221337	16	2.165	0.87
337	2005 WH67	199005	17	1.366	0.70
338	2005 WL160	343864	16.65	1.605	0.10
339	2005 WL35	342219	18.01	0.858	0.34
340	2005 WL69	342703	17.99	0.866	2.02
341	2005 WN149	343718	17.42	1.126	0.99
342	2005 WO138	343554	17.42	1.126	40.40
343	2005 WO31	342151	18.19	0.79	
344	2005 WQ31	342153	17.78	0.954	0.60

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)	Avg Offset (arcsec)
345	2005 WT135	343521	16.29	1.894	1.07
346	2005 WT8	341824	16.07	2.096	0.69
347	2005 WV27	187431	15.7	2.486	0.71
348	2005 WV47	342407	17.45	1.11	0.46
349	2005 WV48	342421	18.58	0.66	1.01
350	2005 WW152	343768	17.99	0.866	47.64
351	2005 WY131	343477	16.79	1.505	4.48
352	2005 XA109	345343	17.03	1.347	0.35
353	2005 XH38	344792	15.5	2.726	0.21
354	2005 XP79	199081	15.4	2.854	0.31
355	2005 XQ15	344538	17.54	1.065	6.43
356	2005 XT61	345032	16.08	2.087	1.05
357	2005 XV69	345125	18.41	0.714	0.93
358	2005 XW19	344583	17.31	1.184	0.62
359	2005 YK98	224620	16.5	1.72	0.27
360	2005 YT18	345670	17.08	1.317	0.28
361	2005 YU17	345659	18.16	0.801	0.15
362	2006 AR	224659	16.9	1.43	0.87
363	2006 BD266	199397	16.3	1.886	0.36
364	2006 BO132	351762	14.06	5.29	0.50
365	2006 CO45	353735	16.09	2.077	1.75
366	2006 DA21	354203	16.35	1.843	0.66
367	2006 DH135	355619	18.23	0.775	5.99
368	2006 DK161	355872	16.92	1.417	0.16
369	2006 DK35	354412	17.13	1.287	0.36
370	2006 DL12	354126	15.45	2.789	0.24
371	2006 DL59	354750	18.04	0.846	0.34
372	2006 DS31	354366	17.56	1.056	2.22
373	2006 EA14	356492	18.08	0.831	0.88
374	2006 HL23	202602	16.2	1.975	
375	2006 JK30	204754	15.9	2.267	0.16
376	2006 KF50	361333	16.67	1.59	0.36
377	2006 KH46	361276	16.88	1.444	0.79
378	2006 KL65	361575	17.18	1.257	43.71
379	2006 KX40	361200	17	1.366	0.56
380	2006 QK23	363554	16.4	1.801	0.83
381	2006 QR78	364244	15.17	3.173	0.61
382	2006 QU42	177984	16.7	1.568	
383	2006 RA46	366116	17.71	0.945	1.12
384	2006 RG88	366845	17.36	1.157	0.72
385	2006 RX55	366287	18.56	0.666	83.57
386	2006 SB132	369555	17.19	1.305	61.19
387	2006 SS235	370837	17.19	1.252	0.51

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)	Avg Offset (arcsec)
388	2006 SW	367564	18.01	0.858	
389	2006 TD109	216190	16	2.165	0.51
390	2006 TM100	374964	18.16	0.801	
391	2006 TW47	374082	16.52	1.704	0.60
392	2006 UK47	376446	17.14	1.281	
393	2006 UM3	375645	17.32	1.179	0.49
394	2006 UN38	376284	16.83	1.477	79.34
395	2006 UY67	181511	17.2	1.246	0.26
396	2006 VK12	380652	18.04	0.846	13.59
397	2006 VP17	380750	17.36	1.246	10.60
398	2006 VP87	381912	16.49	1.728	0.27
399	2006 VR97	382023	16.59	1.642	0.28
400	2006 VT150	221589	17.8	0.945	0.76
401	2006 WC42	216835	17.2	1.246	
402	2006 WJ103	181611	17.2	1.246	0.52
403	2006 WX132	384556	17.36	1.157	
404	2007 DH101	391130	17.05	1.335	0.50
405	2007 EE38	391915	16.8	1.498	0.45
406	2007 EF100	392641	17.59	1.041	0.23
407	2007 EO131	392973	18.04	0.846	0.78
408	2007 EW206	393756	17.72	0.981	1.28
409	2007 GL46	395146	18.24	0.772	32.73
410	2007 HQ90	396504	18.13	0.812	0.52
411	2007 HR19	395768	17.5	1.085	0.31
412	2007 LE28	192213	16.1	2.068	0.78
413	2007 RJ13	204841	16.9	1.43	1.69
414	2007 RL38	399763	16.96	1.391	0.38
415	2007 RM69	400181	17.48	1.095	0.84
416	2007 RU37	399754	17.23	1.229	7.72
417	2007 RU93	400441	17.03	1.347	3.75
418	2007 RV216	401992	16.8	1.498	1.25
419	2007 RX27	399617	17.1	1.305	86.85
420	2007 RZ131	400942	17.12	1.293	
421	2007 TE319	408494	17.57	1.051	13.57
422	2007 TG214	407153	17.52	1.075	1.32
423	2007 TG427	409758	17.23	1.229	
424	2007 TN196	406894	16	2.165	
425	2007 TV237	407524	16	2.165	2.30
426	2007 UK106	411749	17.12	1.293	2.12
427	2007 UQ84	212837	17.5	1.085	0.99
428	2007 VA54	412882	16.17	2.002	1.27
429	2007 VC313	207880	17.2	1.246	0.54
430	2007 VF25	412534	15.01	3.416	0.39

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)	Avg Offset (arcsec)
431	2007 VG220	414685	17.32	1.179	0.47
432	2007 VK288	415374	18.09	0.827	0.69
433	2007 XH30	416770	15.09	3.292	0.29
434	2007 YB8	417127	17.02	1.354	
435	2007 YQ61	417662	16.75	1.533	0.44
436	2008 AM7	417995	17.77	0.958	0.44
437	2008 BL31	419904	17.83	0.932	0.78
438	2008 CF157	422266	16.46	1.752	0.61
439	2008 DB16	225098	17.1	1.305	0.58
440	2008 EG125	425871	16.17	2.002	0.10
441	2008 EU147	189366	16.7	1.568	0.52
442	2008 FA131	428332	12.98	8.699	1.55
443	2008 GG99	429781	17.64	1.017	
444	2008 GQ139	430319	16.7	1.568	0.39
445	2008 HF67	431340	17.26	1.212	3.50
446	2008 JX2	431436	16.81	1.491	1.39
447	2008 KJ	431986	16.43	1.776	0.51
448	2008 KT12	432163	16.5	1.72	2.15
449	2008 OR8	432963	17.34	1.168	0.46
450	2008 PA9	433297	17.88	0.911	0.34
451	2008 PB2	433179	15.06	3.338	2.48
452	2008 PK20	433465	16.75	1.533	0.63
453	2008 RH5	434342	16.28	1.903	0.37
454	2008 SH75	437471	16.07	2.096	0.31
455	2008 SP219	439725	16.47	1.744	0.73
456	2008 ST196	439424	16.66	1.598	0.32
457	2008 TO48	441828	17.82	0.936	0.49
458	2008 UA157	446532	17.44	1.116	33.05
459	2008 UE133	446170	17.85	0.924	
460	2008 UN141	446300	18.04	0.846	0.45
461	2008 UO38	444621	17.26	1.212	26.63
462	2008 UP172	446794	17.18	1.257	0.87
463	2008 UR27	444408	16.26	1.921	79.94
464	2008 UV273	448187	17.87	0.915	0.41
465	2008 VL39	450158	16.76	1.526	0.41
466	2008 VS59	450396	16.25	1.93	0.53
467	2008 WO18	450985	16.72	1.554	2.37
468	2008 WT106	452222	16.21	1.966	0.45
469	2008 XN3	452614	17.68	0.999	1.16
470	2008 YJ161	454944	17.83	0.932	50.40
471	2009 BP106	457172	17.45	1.11	1.00
472	2009 BU71	456639	17.43	1.121	0.89
473	2009 DA126	460597	16.3	1.886	0.22

#	Asteroid Name	Asteroid Id	Absolute Mag	Diameter (m)		Avg Offset (arcsec)
474	2009 DC82	460204	18.86	0.58		
475	2009 DH16	459305	15.6	2.603		
476	2009 DW4	215044	17.9	0.903		0.89
477	2009 FR	461355	26.42	0.018		
478	2009 FY59	462210	13.24	7.718		0.42
479	2009 PJ12	225235	17.4	1.136		0.55
480	2009 RG35	467065	17.58	1.046		0.03
481	2009 RN8	466644	17.49	1.09		0.19
482	2009 RS26	466931	16.71	1.561		0.35
483	2009 SE39	468119	17.51	1.08		1.93
484	2009 SM136	469389	17.22	1.234		0.56
485	2009 SU150	469573	16.77	1.519		0.34
486	2009 SU42	468166	16.94	1.404		
487	2009 UL55	472959	18.24	0.772		75.85
488	2009 WD106	476503	18.64	0.642		1.26
489	2009 WJ159	476593	16.76	1.526		0.46
490	2303 T-3	120430	16.3	1.886		0.47
491	5163 T-2	23384	13.4	7.177	*	1.04
492	6218 P-L	145657	15	3.431		0.81
493	6350 P-L	129375	16.3	1.886		0.39
494	6797 P-L	16306	15.2	3.13		0.23
495	Anchises	1173	8.89	57.211		0.82
496	Casacci	4814	12.7	13.643	*	0.77
497	Epetersen	5350	13.1	5.943	*	0.41
498	Fenska	17951	14.1	3.38	*	1.40
499	Hypnos	14827	18.3	0.751		0.18
500	Leetsungdao	3443	13.3	7.507		0.50
501	Lexcen	18747	14.4	4.524		0.69
502	Mimistrobell	3840	13.2	5.384	*	0.84
503	Offenberger	175259	15.4	2.854		0.92
504	Regidufour	33994	15.6	2.603		0.31
505	Roburnham	4153	12.1	17.337	*	1.44
506	Seishitakeuchi	8575	14.4	4.524		
507	Susil	21229	16	3.945	*	1.99
508	Templehe	22928	15.3	2.989		0.72

APPENDIX F: YARKOVSKY CALCULATIONS, EXAMPLES

Example#1 with a sample of 8 NEO

8 NEO were chosen to test our Yarkorvsky calculation method. The indicated values are justified in the table below. The results were compared with the published one. The differences with the published result are 23 % in average. See below, table 17. (See also: 4.2.3: Comparison with previous calculations).

Table 17 Yarkovsky orbital drift calculations and results for 8 NEO

Asteroid	R	A	Γ	γ	P	ρ	Da/dt_ω	da/dt_n	Σda/dt	#	Drift
Name	M		$J\cdot m^{-2}\cdot K^{-1}\cdot s^{-1/2}$	Deg	hours	$kg\cdot m^3$	10^{-4} x AU/Myr			years	Km
1992 BF	201	0.150	208	150	4.00	2500	-8.49	-0.08	-8.55	52	1869
Golevka	265*	0.150	450*	136	6.03	2700	-10.2	-1.83	-12.04	16	-55
Apollo	750*	0.250	353	162	3.07	2700	-2.83	-0.03	-2.87	78	-685
Bacchus	315*	0.560	249	115	14.90	500	-8.85	-1.19	-10.04	30	-565
Yorp	53	0.150	122	173	0.20	2500	-33.40	-0.01	-33.41	5	-58
Hathor	150*	0.150	185	150	4.00	2500	-9.68	-0.08	-9.76	31	-846
1999 RQ36	204	0.050	600*	170	4.29	1400	-18.58	-0.08	-18.66	7	-54
Geographos	1280*	0.326	437	150	5.22	2500	-1.23	-0.05	-1.28	57	-209

* Radius given, * Thermal inertia given (not calculated from diameter).
For the thermal inertia Γ, when not available, we use the empirical formula $\Gamma = d_0 D^{-\xi}$ described in 4.2.
When not available, average values are used: Rot. Period P= 4 hours, emissivity ε = 0.9, Albedo A = 0.15, Obliquity γ = 150°, density ρ = 2500 $kg\cdot m^{-3}$
The semi-major axis a and eccentricity e values are always taken from NASA / JPL [25].
The radius is calculated from the absolute magnitude when it is not available.

Golevka Radius R, density ρ, albedo A and rotation period P: NASA / JPL [25], Obliquity γ: [14], Thernal inertia [13]

Apollo Radius R, albedo A and rotation period P: NASA / JPL [25], Obliquity γ: [26], Density ρ: J. [27]

Bacchus Rotation period P: NASA / JPL [25], Albedo A, Radius R and density ρ: [28], Obliquity γ: [26]

Yorp Rotation period P: NASA / JPL [25], Obliquity γ: [26] *Hathor* Radius R: NASA / JPL [25]

1999 RQ36 Rotation period P: NASA / JPL [25], Obliquity γ: [26], Density ρ: [29], Albedo A: [30] Thermal inertia Γ: [31]

Geographos Raius R, albedo A and rotation period P: NASA / JPL [25], Obliquity γ: [14], Density ρ: [32]

Example#2 with 6 identified asteroids taken from our sample

6 identified asteroids from our sample were chosen to calculate the Yarkorvsky orbital drift using the same method as the one used for the 8 NEO. The indicated values in the table are justified below. (See table 18).

Table 18 Yarkovsky drift calculations for 6 ast. from the sample

Asteroid	R*	A	Γ	γ	P	P	Da/dt_ω	da/dt_n	Σda/dt	#	Drift
Name	M		$J\cdot m^{-2}\cdot K^{-1}\cdot s^{-1/2}$	Deg	hours	$kg\cdot m^3$	10^{-4} x AU/Myr			years	Km
Hypnos	375	0.150	268	180	4.00	2500	-14.89	0.00	-14.89	6	-8
2009 RG35	523	0.150	305	180	4.00	2500	-8.03	0.00	-8.03	6	-5
2000 TV35	1015	0.150	398	180	4.00	2500	-3.42	0.00	-3.42	6	-2
2005 YU17	401	0.150	275	180	4.00	2500	-10.81	0.00	-10.81	6	-8
2009 WD106	321	0.150	251	180	4.00	2500	-15.69	0.00	-15.59	6	-15
2005 WV48	330	0.150	254	180	4.00	2500	-13.59	0.00	-13.59	6	-10

* Radius calculated from absolute magnitude
For the asteroids in our sample, we chose an obliquity $\gamma = 180^o$ to maximize the Yarkovsky orbital drift.

R: Radius, A: Albedo
da/dt_ω: Diurnal Yarokvsky acceleration
da/dt_n: Seasonal Yarokvsky acceleration
#: Number of years

Γ: Thermal inertia, γ: Obliquity
P: Rotation period, ρ: Density
Σda/dt: Seasonal Yarokvsky acceleration
Drift: Orbital drift in km

LIST OF REFERENCES

[1] The Subaru Prime Focus Camera (Suprime-Cam), A mosaic of ten 2048 x 4096 CCDs, located at the prime focus of Subaru Telescope, covers a 34' x 27' field of view with a pixel scale of 0.20". (http://www.naoj.org/Observing/Instruments/SCam/)

[2] SMOKA Science Archive. The Data Archive of NAOJ (National Astronomical Obervatory of Japan) Telescopes. An online astronomical image archive from the Subaru Telescope. (http://smoka.nao.ac.jp/).

[3] Lowell Observatory, Asteroid Observing Services. List of asteroid orbital parameters and physical properties. The orbital parameters are updated daily. (http://asteroid.lowell.edu/).

[4] FITS World Coordinate System (WCS). The Astronomical Image and Table Format (FITS or Flexible Image Transport System). (http://fits.gsfc.nasa.gov/fits_wcs.html).

[5] The JPL HORIZONS on-line solar system data and ephemeris computation service. (http://ssd.jpl.nasa.gov/?horizons).

[6] SAOImage DS9: Astronomical Data Visualization Application, Joye, W. A.,Mandel, E. Astronomical Data Analysis Software and Systems XII ASP Conference Series, Vol. 295, 2003 H. E. Payne, R. I. Jedrzejewski, and R. N. Hook, eds., p.48. (http://hea-www.harvard.edu/RD/ds9/user/user.pdf)

[7] Microsoft Corporation, "Access 2007 Developer Reference" (2012). (http://msdn.microsoft.com/en-us/library/bb149076.aspx)

[8] Zend, the PHP Company "Php White Papers" (2012). (http://www.zend.com/en/resources/white-papers/).

[9] MySQL Enterprise Edition, "MySQL White Papers" (2012). (http://www.mysql.com/why-mysql/white-papers/).

[10] National Aeronautics and Space Administration (NASA), "Near Earth Object Program". Potentially Hazardous Asteroids.(2012). (http://neo.jpl.nasa.gov/orbits/).

[11] Dustin Lang (Princeton) David W. Hogg (NYU), Astronometry.net.(2012). (http://astrometry.net/).

[12] Wide Field Infrared Survey Explorer (WISE), NASA "MBA Preliminary Albedos and Diameters", (2012). (http://wise2.ipac.caltech.edu/staff/bauer/).

[13] M. Delbo, S. Mouret, M. Havel, M. Carpino, P. Tanga, D. Hestroffer, F. Mignard, and A. Morbidelli, " Thermal inertia of near-Earth asteroids and implications for the magnitude of the Yarkovsky", doi:10.1016/j.icarus.2007.03.007, Submitted to ICARUS: November 26, 2006, Revised: March 2, 2007. Accepted: March 5, 2007 (http://arxiv.org/ftp/arxiv/papers/0704/0704.1915.pdf) page 31.

[14] D. Vokrouhlick´y, A. Milani, S. R. Chesley, "Yarkovsky Effect on Small Near-Earth Asteroids: Mathematical Formulation and Examples", Icarus 148, 118–138 (2000), Volume 148, Issue 1, November 2000, Pages 118–138 (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.31.6837).

[15] Chesley, S.R., Vokrouhlicky D., Ostro, S.J., and 5 colleagues, "Direct estimation of Yarkovsky acceleration on near-Earth asteroids", Presented at ACM08 meeting, Baltimore (2008). (http://astro.mff.cuni.cz/davok/papers/Chesley_etal_ACM08.pdf)

[16] Chesley, Steven R.; Vokrouhlicky, D., "Searching for Yarkovsky among the Near-Earth Asteroids", presented at American Astronomical Society, DDA meeting #39, #2.04 (05/2008). (http://adsabs.harvard.edu//abs/2008DDA....39.0204C)

[17] Chesley, Steven R.; and Al., "Direct Detection of the Yarkovsky Effect via Radar Ranging to Asteroid 6489 Golevka", Science Mag., Vol. 302 no. 5651 pp. 1739-1742 (2003) (http://www.sciencemag.org/content/302/5651/1739)

[18] D. Vokrouhlický, S. R. Chesley and R.D. Matson, "Orbital Identification for Asteroid 152563 (1992 BF) Through the Yarkovsky effect", The Astronomical Journal 135 2336. (2008) (http://iopscience.iop.org/1538-3881/135/6/2336/)

[19] David Parry Rubincam "Yarkovsky thermal drag on small asteroids and Mars-Earth delivery", Journal of Geophysical Research, Vol. 103, NO. E1, PP. 1725-1732, 1998 (http://www.agu.org/pubs/crossref/1998/97JE03034.shtml)

[20] David Parry Rubincam "Asteroid orbit evolution due to thermal drag", Journal of Geophysical Research (ISSN 0148-0227), vol. 100, no. E1, p. 1585-1594, 01/1995 (http://adsabs.harvard.edu//abs/1995JGR...100.1585R)

[21] Vokrouhlický, D.; Nesvorný, D.; Dones, L.; Bottke, W. F. "Thermal forces on planetary ring particles: application to the main system of Saturn", Astronomy and Astrophysics, Volume 471, Issue 2, August IV 2007, pp.717-730 (2007) (http://adsabs.harvard.edu//abs/2007A%26A...471..717V)

[22] Skoglöv, E. "The influence on the spin vectors of asteroids from the Yarkovsky effect", Astronomy and Astrophysics, v.393, p.673-683 (2002) (http://adsabs.harvard.edu//abs/2002A%26A...393..673S)

[23] Bottke, W. F., Jr.; Vokrouhlický, D.; Rubincam, D. P.; Broz, M. "The Effect of Yarkovsky Thermal Forces on the Dynamical Evolution of Asteroids and Meteoroids", Asteroids III, W. F. Bottke Jr., A. Cellino, P. Paolicchi, and R. P. Binzel (eds), University of Arizona Press, Tucson, p.395-408 (2002) (http://www.lpi.usra.edu/books/AsteroidsIII/pdf/3014.pdf)

[24] Peter Duffet Smith, Second Edition. "Practical astronomy with your calculator" Cambridge University Press 1979, 1981.

[25] Jet Propulsion Laboratory, California Institute of Technology "JPL Small-Body Database Browser" (2012). (http://ssd.jpl.nasa.gov/?bodies).

[26] William F. Bottke, Jr., David Vokrouhlick´y, "Yarkovsky and YORP effects" Scholarpedia, "The peer reviewed open-access encyclopedia" (2012). (http://www.scholarpedia.org/article/Yarkovsky_and_YORP_effects)

[27] J. ˇDurech, D. Vokrouhlický, M. Kaasalainen, P. Weissman, S. C. Lowry, E. Beshore, D. Higgins, Y. N. Krugly,V. G. Shevchenko, N. M. Gaftonyuk, Y.-J. Choi,R. A. Kowalski, S. Larson, B. D.Warner, A. L. Marshalkina,M. A. Ibrahimov, I. E. Molotov, T. Michałowski, and K. Kitazato, "New photometric observations of asteroids (1862) Apollo and (25143) Itokawa – an analysis of YORP effect", A&A 488, 345–350 (2008) (http://adsabs.harvard.edu/abs/2008A%26A...488..345D) Page 346.

[28] Lance A. M. Benner, R. Scott Hudson "Radar Observations of Asteroid 2063 Bacchus" Icarus 139, 309–327 (1999). (http://echo.jpl.nasa.gov/asteroids/benner.etal.1999.bacchus.pdf).

[29] J.P. Emery, Y.R. Fernández, M.S. Kelley, C. Hergenrother, J. Ziffer, D.S. Lauretta, M.J. Drake, H. Campins. "Thermophysical Characterization of Potential Spacecraft Target (101955) 1999 RQ36" 41st Lunar and Planetary Science Conference, 03/01/2010 Woodlands, Texas. LPI Contribution No. 1533, p.2282 (http://www.lpi.usra.edu/meetings/lpsc2010/pdf/2282.pdf).

[30] Andrea Milani, Steven R. Chesley, Maria Eugenia Sansaturio, Fabrizio Bernardi,1, Giovanni B. Valsecchi, Oscar Arratia "Long term impact risk for (101955) 1999 RQ36", arXiv.org > astro-ph > arXiv:0901.3631v1 (2009) (http://arxiv.org/PS_cache/arxiv/pdf/0901/0901.3631v1.pdf)

[31] D. S. Lauretta1, M. J. Drake1, R. P. Binzel, H. Campins. "Asteroid (101955) 1999RQ36: Optimum target for an asteroid sample return mission". 73rd Annual Meteoritical Society Meeting (2010) (http://www.lpi.usra.edu/meetings/metsoc2010/pdf/5153.pdf).

[32] J. ˇDurech, D. Vokrouhlický, M. Kaasalainen, D. Higgins, Yu. N. Krugly, N. M. Gaftonyuk, V. G. Shevchenko,V. G. Chiorny, H. Hamanowa, H. Hamanowa, V. Reddy, and R. R. Dyvig "Detection of the YORP effect in asteroid (1620) Geographos", Astronomy & Astrophysics (2008): A&A 489, L25–L28 (2008)157-191 DOI: 10.1051/0004-6361:200810672 (http://adsabs.harvard.edu/abs/2008A%26A...489L..25D)

[33] Satoshi Miyazaki, Yutaka Komiyama, Maki Sekiguchi, Sadanori Okamura "Subaru, Prime Focus Camera - Suprime-Cam", Astrophysics (astro-ph),2008. Astronomical Society of Japan., arXiv:astro-ph/0211006v2 (2002) (http://arxiv.org/pdf/astro-ph/0211006v2.pdf)

[34] Dustin Lang, David W. Hogg, Keir Mierle, Michael Blanton, Sam Roweis "Astrometry.net: Blind astrometric calibration of arbitrary astronomical images ", 2010 The Astronomical Journal 139 1782 (http://arxiv.org/pdf/0910.2233v1.pdf)

[35] Delbo', Marco; Tanga, Paolo "Thermal inertia of main belt asteroids smaller than 100 km from IRAS data", Planetary and Space Science, Volume 57, Issue 2, p. 259-265, 2009 (http://arxiv.org/pdf/0808.0869v1.pdf)